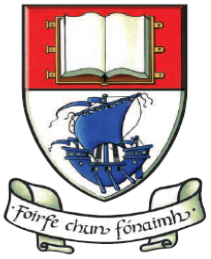


Waterford Institute *of* Technology

Machine Learning based Stochastic Techniques for Collaborative Privacy in Social Recommender Services

Thesis Submitted in partial fulfilment of the requirements for the award of *Doctor of Philosophy*

Ahmed Mohamed Khamis Elmesiry, M.Sc. (No: W20038101)

Department of Computing, Mathematics and Physics
Waterford Institute of Technology

Supervisor: Dr. Dmitri Botvich
Mentor Supervisor: Dr. Micheal O Foghlu

Submitted to Waterford Institute of Technology, September 2014

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy, is entirely my own work and has not been taken from the work of others save to the extent that such work has been cited and acknowledged within the text of my work.

Signed……Ahmed Mohamed Elmesiry…….. No.20038101

Date: September 2014

Abstract of Thesis:

A variety of online social services have been developed over the last decade. They have all had a profound effect on today's society. With the emergence of Web 2.0 and the popularity of social media, there has been a growing demand to provide services supporting social network platforms. New services are constantly being developed, where an increasing volume of personal data is being processed in return for personally tailored services. This result in creating mature systems to satisfy users' needs is referred to as social recommender services that create an inevitable trend driven by mutual benefits. The penetration of these services has been relatively slow recently since there still exists different viewpoints regarding the exploitation (and thus potential) of those services amongst researchers and users.

One major concern regarding their adoption lies in privacy considerations of the users while using these services. With the increasing amount of personal data that users distribute over different services, there is an increased probability for identity fraud, profiling and linkability attacks, that not only poses a threat to those people's personal dignity and affects different aspects of their lives, but also to societies as a whole. Furthermore, the growing adoption of remote data processing and storage for these services pose a number of privacy issues. As a result, users of those services have shown an increasing concern for exposing their personal data to untrusted entities so as to receive value-added services [4]. They need to realize full control over their sensitive data collected by these services and cannot accept that their data might be fully accessible to an external third party. In most cases, this can forestall these users from fully embracing these social recommender services. We argue that current social recommender services presents a number of specific privacy issues and problems inherited from the outsourcing architecture which represent the backbone of those services. This thesis supports the need for mature and extensible solutions that aid in retaining control for the users over their personal data.

Most of the "privacy-concerned" systems that have been developed so far, are either based on a trusted third-party model or on some generalized architecture. Moreover, other systems address this problem with techniques to protect the processing of data stored on untrusted providers. The current techniques and tools for protecting the privacy of users'

personal data on the Internet, is referred to as Privacy Enhancing Technologies (PETs). Various PETs are being designed and developed. However, they are usually seen by ordinary users as complicated and disruptive of their primary tasks. In order to address these issues, this thesis presents a novel approach where sensitive data has two copies, a concealed version that is located on the services side and a plain version on the client side. Our approach for enhancing the users' privacy is to deploy a middleware on the client side where his/her data can be either kept private, or released in a locally concealed form. The latter implies that data is shared in a private manner after concealing it on the user's side using local concealment techniques. We built a middleware that takes into consideration the social side for these recommender services. This middleware can be utilized for these social recommender services to facilitate access to a wealth of users' data in a manner that preserves privacy. Our aim is not only to limit and prevent the disclosure of sensitive data but also to preserve the usefulness of data as much as possible to be only effective for the required computation.

This thesis focuses primarily on social recommender services which are of great interest. On the one hand they lay the groundwork for new innovative applications but on the other hand they pose numerous unique challenges to privacy. We studied the privacy problem faced by people in sharing their profiles' preferences within various scenarios of social recommender services. We proposed and developed a collaborative privacy approach for preserving users' profile privacy and we have applied this approach to representative scenarios: (I.) Recommender service for IPTV content providers; (II.) Data Mash-up services for IPTV recommender services and (III.) Community discovery and recommendation services for implicit social groups (conference organization and university campus). Location based recommendation services, mobile jukebox content recommender services, and pervasive healthcare services were studied and enhanced as well in order to show the applicability of our approach. We discussed how our approach could handle the privacy problem in these scenarios. In addition, the proposed collaborative privacy framework was developed as a middleware that hosts a set of components to execute a two stage concealment process with novel stochastic techniques. Each stage in the two stage concealment process is carried out by completely different parties depending on their role in the coalition. The proposed middleware as well as the set of components and techniques

that is employed in its implementation, permit the end-users to control the privacy of their released data while interacting with social recommender services. This kind of approach is quite flexible and can easily be adopted in conventional social recommender services because it is executed on the user side and takes advantage of the social structure that is offered by the online social service without the need for significant modifications at the service provider side. The attained accuracy and privacy levels for the data concealed using the proposed stochastic techniques in different scenarios were evaluated. Moreover, attacks on such concealed data were presented to demonstrate the stability of our proposed techniques against such attacks. Finally, we applied off-the-shelf recommendation techniques to make referrals as a show case. Therefore, the experimental results show that the proposed approach obtains accurate results similar to unsecured services, while at the same time meeting users' privacy concerns.

Acknowledgements

I want to thank many people who contributed to the final result of this work. First, I am deeply grateful to my supervisor, Dr. Dmitri Botvich. His insight, guidance, advice and help were invaluable in shaping this thesis possible. I would like to thank him for the countless hours we spent together discussing the ideas and concepts of the proposed framework. I also thank him for all the words of encouragement that helped me to safely pass along all the tough moments of despair and to solve even the most difficult problems that we faced when I started this PhD research. I am also thankful to Dr. Micheal O Foghlu. He taught me critical thinking and technical writing skills, which laid a solid foundation for my future research.

In this tiny space, how does one thank someone who has always loved you since the day that you were born and, whose daily prayers for you to have all the success and happiness in your life. There are two people who come to my mind when I ask myself this question. They are my late father and my mother. Thank you, father for all what you have done for me. If it was not for your friendship and understanding, I certainly would not have had the courage to strive and achieve a fraction of what I have managed. You are, and always will be, my unrivalled number one friend. You are no longer with us in a physical sense, but you live on each day in my heart. A huge hug to my lovely mother whose love and belief in me has helped me achieve what I have. I thank you for your never ending love to me and for believing that a formal education would be the best direction that I could take in life. Thank you for all of the sacrifices you made for me in your life, so that I could reach for the stars today. I also want to give special thanks to my sister who is the most compassionate and kindest person in the world. I wish for you to have all of the success in your life. I also owe many thanks to my fiancée, Mirela, for her patience and by supporting my research faithfully all the time. I must say that her presence and warm words were not only helpful, but also provided encouragement and motivation for me. You stood by my side during this year and had to endure the painful moments when I was finalizing this thesis. For that, you have my gratitude and love forever. Thanks to my family who has endured my long absences for the PhD throughout the years. Their support and love made my life enjoyable

even in those stressful times. You all have been and will always be the reasons that I strive for excellence.

Last but not least I am obliged to my colleagues, namely Dr. Brendan Jennings, Dr. Garjaruban Kandavanam, Dr. Julien Mineraud, Dr. Leigh Griffin, Dr. Ray Carrol, Dr. Stepan Ivanov, Mrs. Annie Ibrahim, Mr. Bernard Butler, Mr. Eric Robson, Mr. Jason Barron, Mrs. Armita Afsharinejad , Mr. Biao Xu, Mrs. Zohra Boudjemil, Mr. Shahidul Hoque, Mr. Mike White, Mr. Mansoor Ahmad, Mr. Hisain Elshaafi, Mr. Michael Taynnan Barros, Mr. Brian Meskill, Mr. Siblee Islam and Mr. Mohamed Adel. Without them, the life of my PhD studies would lose a lot of fun. Moreover, I extend my thanks to all of the TSSG for making it such a friendly and inspiring workplace for positive interactions and research perspectives.

Finally, my study and research at the Waterford Institute of Technology would have been impossible without generous financial support from TSSG in the form of a research grant.

Multitudes of thank yous!

Table of Contents

This thesis consists of an introductory part and the following fifteen original peer-reviewed publications, as well as one additional article currently in a peer-review process for journal publications (Article XVI), presented in chronological order, and grouped by research themes and reprinted in the appendices at the end pf the thesis.

Part I: Thesis Introduction

Part II: Included Papers

List of the Author's Publications

Journal Articles

1. **Martín Serrano, Ahmed Elmesiry, Mícheál Ó Foghlú, Willie Donnelly, Cristiano Storni and Mikael Fernström.** Pervasive Computing Support in the Transition towards Personalized Health Systems **Published in the International Journal on E-Health and Medical Communications (IJEHMC)- Volume 2, Issue 3, July 2011.**
2. **Ahmed M. Elmisery, Dimtri Botvich**. Enhanced Middleware for Collaborative Privacy in IPTV Recommender Recommender Services **Published in Journal of Convergence (JOC)-2011-vol.2 issue 2**
3. **Ahmed M. Elmisery, Dimtri Botvich**. Privacy Aware Recommender Service using Multi-agent Middleware- an IPTV Network Scenario **Published in the International Journal of Computing and Informatics (Informatica)- Volume 36, Issue 1, March 2012**
4. **Ahmed M. Elmisery, Dimtri Botvich**. Multi-Agent Based Middleware for Protecting Privacy in IPTV Content Recommender **Published in the International Journal of Multimedia Tools and Applications- March 2012-vol.64 issue 2**
5. **Ahmed M. Elmisery.** Private Personalized Social Recommendations in an IPTV System. **(Accepted for Publication in New Review of Hypermedia and Multimedia - Taylor & Francis- June 2014- vol.20 issue 2.**
6. **Ahmed M. Elmisery, Dimtri Botvich**. Privacy Enhanced Middleware for Collaborative Virtual Sub-Community Discovery in Implicit Social Groups. **(Accepted for Publication in Electronic Commerce Research)**.
7. **Ahmed M. Elmisery, Dimtri Botvich**. A Distributed Collaborative Platform for Personal Health Profiles in Patient-Driven Health Social Network. **(Submitted 2014)**.
8. **Ahmed M. Elmisery, Dimtri Botvich**. Collaborative Privacy Framework for Minimizing Privacy Risks In Social Recommender Services **Published in the International Journal of Multimedia Tools and Applications- September 2014**.
9. **Ahmed M. Elmisery, S Rho, Dimtri Botvich**. Holistic Collaborative Privacy Solution for User Privacy in Mobile Jukebox Recommender Services **Published in the International Journal of Platform Technology-March 2014- vol.2issue 1.**

Conference Papers

10. **Ahmed M. Elmisery** Advanced Firewall Architecture for Virtual Private Networks. **In Proceedings, 7th Alazhar International Conference on Computer And Engineering, (AEIC '2003) ,Cairo, May 2003.**

11. **Ahmed M. Elmisery** An English-To-Arabic Translation Agent. **In Proceedings, 1st International Computer Engineering Conference New Technologies for the Information Society, (ICENCO'2004) ,Cairo, December 2004.**
12. **Ahmed M. Elmisery, Mohamed M. Kouta, Mohammed M. Abou Rizka.** Securing Mobile Agents in Hostile Environment. **In Proceedings, 4th International Conference on Computer Theory and Applications, (INFOS'2006). Cairo, March 2006.**
13. **Ahmed M. Elmisery, Mohamed M. Kouta, Mohammed M. Abou Rizka.** I-AGENT : A System For Multi-agent Interoperability.**In Proceedings, 4th International Conference on Computer Theory and Applications, (INFOS'2006). Cairo, March 2006**
14. **Mohamed M. Kouta, Mohammed M. Abou Rizka , Ahmed M. Elmisery.** Secure e-Payment using Multi-agent Architecture. **In Proceedings, International Workshop on Engineering Semantic Agent Systems (ESAS'2006); In conjunction with 30th IEEE Annual International Computer Software and Applications Chicago, USA, September 2006.**
15. **Ahmed M. Elmisery, Mohamed M. Kouta, Mohammed M. Abou Rizka.** Hybrid Model for Agent-P2P ecommerce Systems. **In Proceedings, 16th International Conference on Computer Theory and Applications, (ICCTA'2006). Alexandria, September 2006.**
16. **Ahmed M. Elmisery, Mohamed M. Kouta, Mohammed M. Abou Rizka.** APES: An Agent Peer To Peer eCommerce System. **In Proceedings, 16th International Conference on Computer Theory and Applications, (ICCTA'2006). Alexandria, September 2006.**
17. **Ahmed M. Elmisery, Mohamed M. Kouta, Mohammed M. Abou Rizka.** Rule Based Negotiation Model for APES. **In Proceedings, 16th International Conference on Computer Theory and Applications, (ICCTA'2006). Alexandria, September 2006.**
18. **Ahmed M. Elmisery, Mohamed M. Kouta, Mohammed M. Abou Rizka.** Security Framework for Agent Based E-payment Systems. **In Proceedings, 17th International Conference on Computer Theory and Applications, (ICCTA'2007). Alexandria, September 2007.**
19. **Ahmed M. Elmisery, F.I.Mohammed.** Using Agent Technology for Personalized E-learning **In Proceedings, 17th International Conference on Computer Theory and Applications, (ICCTA'2007). Alexandria, September 2007.**
20. **H.Fu, Ahmed M. Elmisery.** A New Feature Weighted Fuzzy C-means Clustering Algorithm **In Proceedings, IADIS European Conference on Data Mining 2009, Algarve, Portugal, June 2009.**
21. **Ahmed M. Elmisery, H.Fu.** Privacy Preserving Distributed Clustering Of HealthCare Data Using Cryptography Protocols. **In Proceedings, The 5th IEEE International Workshop on Security, Trust, and Privacy for Software Applications (STPSA 2010); In conjunction with 34th IEEE Annual International Computer Software and Applications Seoul, South Korea, July 2010.**

22. **Ahmed M. Elmisery, D.Botvich.** Privacy for Location Based Recommender Services-Case Study in Waze. **Proc. 1st Summer School on Mobility, Data Mining, and Privacy, Rhodos Island, Greece 2010.**
23. **Ahmed M. Elmisery, Dimtri Botvich.** Private Recommendation Service for IPTV System. **In Proceedings, 12th IFIP/IEEE IM 2011, Dublin, Ireland, May 2011.**
24. **Ahmed M. Elmisery, Dimtri Botvich.** An Agent Based Middleware for Privacy Aware Recommender Systems in IPTV Networks.**In Proc.3rd International Conference on Intelligent Decision Technologies . Springer Berlin Heidelberg, 2011.**
25. **Ahmed M. Elmisery, Dimtri Botvich.** Agent Based Middleware for Private Data Mashup in IPTV Recommender Services. **In Proc. 16th IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD 2011) . IEEE computer society, June 2011.**
26. **Ahmed M. Elmisery, Dimtri Botvich.** Privacy Aware Recommender Service for IPTV Networks.**In Proc. 5th FTRA/IEEE International Conference on Multimedia and Ubiquitous Engineering (MUE 2011) . IEEE computer society, June 2011.**
27. **Ahmed M. Elmisery, Dimtri Botvich.** Agent Based Middleware for Maintaining User Privacy in IPTV Recommender Services**. In Proc. 3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems (MOBISEC 2011) . Springer Berlin Heidelberg, May 2011.**
28. **Ahmed Elmesiry , Martín Serrano, Dimtri Botvich.** Diagnosis Support on cardio-Vascular Signal Monitoring by using Cluster Computing,**In Proc.3rd International Conference on Intelligent Decision Technologies . Springer Berlin Heidelberg, 2011.**
29. **Ahmed M. Elmisery, Dimtri Botvich.** Privacy Aware Obfuscation Middleware for Mobile Jukebox Recommender Services. **In Proc.11th IFIP Conference on e-Business, e-Service, e-Society (I3E 2011)**
30. **Ahmed M. Elmisery, Kevin Doolin, Dmitri Botvich.** Privacy Aware Community based Recommender Service for Conferences Attendees. **In Proc.16th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems. IOS Press, San Sebastian, Spain (KES 2012)**
31. **Ahmed M. Elmisery, Dmitri Botvich.** Maintaining Privacy of Data Outsourced to a Cloud based Recommender Service - Privacy by Deign Approach. **In Proc. 2nd Summer School on Privacy aware social mining workshop, Leysin, Switzerland 2012.**
32. **Ahmed M. Elmisery, Kevin Doolin, Ioanna Roussaki, Dmitri Botvich.** Enhanced Middleware for Collaborative Privacy in Community based Recommendations Services. **In Proc. 4th FTRA International Conference on Computer Science and its Applications (CSA-12), Jeju, Korea**
33. **Ahmed M. Elmisery, Dmitri Botvich.** Privacy of Data Outsourced to a Cloud based Recommender Service. **Proc. 2nd Summer School on Privacy aware social mining, Leysin, Switzerland 2012.**

Chapter 1

Introduction

In this chapter, the background, motivation, challenges, scope, and contributions of the research undertaken in this thesis will be addressed. First, the background will be presented in Section 1.1, which will be followed by the motivation of this thesis in Section 1.2. The research challenges and scope of this thesis will be presented in Sections 1.3 and 1.4 respectively. Sections 1.5 and 1.6 will describe the research questions and contributions of this thesis. Finally, Section 1.7 will illustrate the organization of this thesis.

1.1 Background

The invention of the internet and social networks has had a major impact on society and the way we are living. The majority of people access the internet either via desktop computers or powerful mobile devices. There has also been a migration of services from the normal paper-based world to the electronic world and the rapid deployment of the novel types of these electronic services, within which a growing amount of personal data is being collected by service providers in return for highly personalized services. This collected data is often crammed with personal sensitive data (e.g., medical data, consumption profiles, and monetary data). This phenomenon has been further aggravated by the increases in computational and storage performance and the sharp decline in technology costs. This reality is well delineated by the author in [5]. For instance, he says:

> "*Small details that were once captured in dim memories or fading scraps of paper are now preserved forever in the digital minds of computers, in vast databases with fertile fields of personal data. Our wallets are stuffed with ATM cards, calling cards, frequent shopper cards, and credit cards—all of which can be used to record where we*

are and what we do. Every day, rivulets of information stream into electric brains to be sifted, sorted, rearranged, and combined in hundreds of different ways. Digital technology enables the preservation of the minutia of our everyday comings and goings, of our likes and dislikes, of who we are and what we own. It is ever more possible to create an electronic collage that covers much of a person's life—a life captured in records, a digital person composed in the collective computer networks of the world".

One common example of those new electronic services are location based services, where the user reveals his/her current location to the service provider in return for services such as the location of nearby friends or traffic conditions. Another important topic is that the recommender services which fight against the bombardment of information, have the challenging task of presenting users with results that are of interest to them and filtering out tangential parts. Thus, the more a user reveals of his/her personal data, the more substantive the recommendations are. However, revealing personal data goes against the user's requirement of privacy. Furthermore, incidents of data breaches are unfortunately quite common and a reasonable concern is that many of them did not gain abundant attention within the media. One example of such incidents is the NYSE case. Choicepoint is a data aggregation company that acted as a private intelligence service to the government and private sector. In 2004, Choicepoint sold the private records of customers to a group of criminals [6]. Another incident occurred in 2008. A Deutsche Telekom company (T-Mobile) lost 17 million subscribers' personal data when a computer disc containing these data was lost [7]. In September 2006, AOL released a dataset with search query-log data containing about 21 million web queries collected from about 650 thousand users over 3 months. This data was anonymized to protect user privacy. Wherever a real IP address appeared, it had been replaced with a random ID number. Shortly after the release, the first 'anonymous' user had been identified from the log data. In particular, the user given the ID

4417749 in AOL's query-log was identified as the 62-year-old Thelma [8]. All these cases indicate that personal data is a crucial, valuable resource that needs to be protected in order to ensure the individual's privacy.

Privacy is an elusive concept that is difficult to define. It is not an entirely technical subject but it is connected to aspects of legislation, service providers' policies, and social norms. Privacy is an adjustable notion that depends on the users' perception of risk and profit. Some users agree to reveal their personal information if they are given incentives in return. This can be in the form of a discount coupon, accurate referrals, or personalized content. However, when something is considered private to the user, it usually means there is something within them that is considered inherently personally sensitive to avoid discrimination, personal embarrassment, or harm to their professional reputations. The degree to which private information is exposed thus depends on how the public will receive this information, which differs between situations and over time. We have found several viewpoints covering the extent of privacy definition in our research. However, we need to cover some of the main viewpoints which can be fundamentals for proceeding with this work. Warren and Brandeis defined privacy in 1890 as "the right to be let alone" [9]. In the age of information and communication technologies, Westin [10] states that privacy can be further divided into informational and spatial privacy. Informational privacy denotes that the user can control how, when, and to what extent information about him/her is released to others. This is often associated with any personal information such as name, age, phone number, or e-mail. Further, spatial privacy denotes that the user can exercise control over what information is presented to his/her senses. The research in [11] conceptualizes privacy as the "selective control of access to the self" regulated as dialectic and dynamic processes that embrace multi-mechanistic optimizing behaviours. The author in [12] argues that privacy is neither rule based nor static. Instead, "a fine and shifting line between privacy and publicity exists, and depends on the social context, intention, and the fine-grained

coordination between action and the disclosure of that action". Clearly, the concept of privacy is often more complicated than realized and still unclear as it varies based on the various contexts of provided services [13]. A notable exception is the work presented in [14], in which privacy was defined as permitting services to extract a valid knowledge without learning the underlying users' personal data. At this point, each of the privacy enhancing technologies has its own privacy definition. Our primary concern about PETs is that services are analysed for the side effects they incur due to applying privacy enhancing techniques. Therefore, our definition of privacy is close to previous definitions that encompass the dual goal of meeting privacy requirements and providing valid results. Our definition emphasizes the dilemma of balancing privacy and accuracy.

The gradual loss of privacy in today's digital life is due to the fact that "formidable dossiers" are being built regarding individual users. This is due in part to the spread of electronic services and also the ubiquity of their use with the possibility to pool and link this data with other service providers and/or governments [15]. There is also the spread of unfair data collection practices and the lack of regulations. For example, service providers frequently collect data regarding their customers, presumably to better serve them. However, a significant part of the data that is typically collected is not essential to the service being offered, or to the completion of the services it was presumably released for. Gathering such unnecessary data can be seen as a privacy threat, and storing it exposes the customer to further unavoidable risks. The potential abuses from unscrupulous merchants that were identified have caused increasing attention among the general public and within the media in recent years. Numerous surveys pointed out the users need for privacy [16]. Regarding the results of the surveys, the report on personal privacy in [17] indicates that over 81% of the people in the survey were willing to provide information as long as their privacy was guaranteed. A subsequent study in [18] used customer surveys to investigate the impact that personalized systems have on their privacy preferences. Customers are

willing to share the personal information with the service providers once they are allowed to select, edit, and delete their data. For instance, 69% of customers wanted to have control of their data and around 90% were concerned about the sharing of their data by service providers for a purpose completely different from what it was released for. These surveys demonstrate the great importance that users placed in retaining control over their personal data even when they are on the service provider's side. The implementation of privacy aware frameworks for protecting the user's personal data is a step in this direction.

Privacy violations are prohibited in many countries. However, there is an absence of effective methods to enforce the law. This problem is exacerbated once information is used about individuals without their knowledge. As they should, if the customer has a proof that his/her privacy has been violated by the merchant, he could complain to the proper authorities, so that justice might be served. However, no amount of "justice" can fully restore his/her privacy. Two common methods can be utilized for guaranteeing the privacy including technological, and legislation solutions. The former approach refers to technical methods and tools that integrated into systems or networks in order to reduce the collection of accurate personal data. Such methods and tools are referred to as privacy enhancing technologies (PETs). One example of a PET, which is discussed in this thesis, is the two stage concealment process that aims to control the amount of information the users reveal in the initial contact, eliminates the necessity to release personal data in raw form and permits the users to act anonymously. Privacy legislation refers to data protection legislation restricting the gathering and usage of private personal data by data processors. Two examples of privacy guidelines are the EU Directives 95/46/EC [19] and 2002/58/EC [20]. Despite the fact that several nations have developed privacy protection laws and regulations to guard against secret use of personal information, the present laws and their conceptual foundations have become outdated because of the continuous changes in technology [21]. As a result, these personal data reside on databases of service providers,

largely beyond the control of existing privacy laws, leading to potential privacy invasion on a scale never before possible. It is commonly believed that privacy is most successfully protected by a holistic solution that combines both technological and legislative efforts.

1.2 Motivation

Recent decades have seen tremendous growth in the scale and complexity of data analysis systems. An example is recommender systems which are typically in the form of filtering data in the quest for useful actionable knowledge to accommodate the differences between individuals [22]. It has tremendous business value since most organizations have an interest in filtering customer records for improving and refining their marketing campaigns. It has also given rise to new business opportunities. The success of web search engines, such as Google, is an excellent example of this phenomenon, with a market capitalization of around 200 billion USD and earnings 98% of its revenue from its search business. Web search engines are aimed toward helping users find their required information quickly. To achieve this goal, search engines are required to process web data so that once the user enters a search query, a ranked list of relevant results is displayed in a timely manner.

Both of the above mentioned domains require processing of extremely large data sets for identification of fine grained patterns. This pattern discovery is vital for business organizations because it helps them in identifying potential customer targets for specific products or services [23]. It is also required for web search engines to determine the similarity among various documents based on the co-occurrence pattern of various keywords and terms [24]. Traditional clustering techniques have shown to be inadequate for finding fine grained patterns [25] in a distributed environment. Sometimes, we do not only want to identify object groups based on their features on a specific database on one site, but we also want to know which groups are common across different sites based on corresponding objects. The need for distributed clustering is well known in the case of text

mining, healthcare data analysis, and social recommender services [26]. Recent studies [27] indicate that distributed clustering is also very effective in recommender services. Traditional clustering techniques such as *K*-means and expectation maximization (EM) clustering are unable to scale and perform accurate distributed clustering.

To meet these challenges of distributed clustering, a variety of techniques have been proposed [28] which have been popularly referred to as distributed data clustering. Throughout this thesis, the term grouping will be used to refer to clustering and items to refer to contents or products that are offered by recommender systems such as movies, songs, and nearest points of interest or special interest communities. The need for an efficient distributed clustering solution will be described first in Section 1.2.1. Finally, Section 1.2.2 will discuss the necessity for preserving users' privacy in social recommendation services.

1.2.1 An Efficient Distributed Clustering Solution

New business models rely on the collaboration between different organizations. This collaboration takes the form of building machine learning models in their shared data in order to achieve mutual benefits. One of the most used machine learning models is distributed clustering, which involves finding optimal clusters in their data, then utilizing these clusters for target marketing, fraud detection, customer segmentation, and service personalization. However, there are two obstacles toward achieving this collaboration (1) the challenges involved in finding common clusters are intrinsically more complex than simple clustering and most distributed data clustering techniques are not efficient for practical deployment [29] and (2) the privacy regulations of each organization restrict data sharing with external parties. Amid these efficiency and privacy considerations, there is a strong need for efficient distributed clustering solutions which can be deployed in large scale practical systems. We intend to provide a solution to this downside in this thesis.

Amid intensive use of machine learning clustering analysis for pattern discovery, there have been a number of privacy issues raised, specifically in healthcare and social recommender services. There has been some progress made in privacy enhancing technologies but most of those approaches are not practically viable because of the increased computational cost and reduced utility of data. In this thesis, we intend to provide solutions for these challenges. Specifically, we attempt to reduce the gap between theoretical secure multiparty computation protocols and requirements for practical privacy enhancing solutions.

As mentioned above, one of the main topics of this thesis is an efficient distributed clustering algorithm for fine grained pattern discovery. We termed the proposed algorithm as distributed local clustering (DLC), which was designed to produce accurate clusters with arbitrary shapes, sizes, and densities over vertically partitioned databases. Moreover, it can be utilized in privacy preserving scenarios as the model building retrieves the original statistical properties without the need to collect the entire original user's data from each site. Traditional clustering approaches detect clusters based on similarity over the entire set of attributes. This requires a single site to be in charge of collecting all datasets from all sites, which is not an applicable case in a distributed environment. DLC employs various objective functions within two consecutive steps in order to detect global clusters across all sites that provide an optimal solution for these functions. The first step is used to detect local dense regions or clusters on each site independently. Thereafter, the final step is used to create global dense regions or clusters by merging all of the discovered local dense regions.

In order to demonstrate the effectiveness of the proposed distributed clustering algorithm in solving sensible real world problems, we have focused on personalized healthcare systems. A personalized healthcare system (pHealth) is a managing environment for various healthcare applications running within the same computing environment. These systems

offer massive healthcare data analysis capabilities, distributed data storage and health communication networks. An example of such an application is personalized medical support for cardiovascular monitoring which is a monitoring application for the remote diagnosis of cardiovascular signals hosted on a cluster computing environment. It performs an automated assignment for a patient to a physiological condition using no prior knowledge about disease states. We intended to show that medical diagnosis of cardiovascular signals for this scenario can be achieved efficiently through the proposed distributed clustering algorithm (DLC). The algorithm collects different statistics from the cardiac signals and uses these statistics to build a distributed clustering model automatically. The resulting model can be used for diagnostic purposes of various cardiac signals.

1.2.2 Preserving User Privacy in Social Recommender Services

In order to respond to the privacy challenges, we have focused on building a collaborative privacy framework adopting a two stage concealment process which utilizes the social nature of the online social service. The current randomization techniques can be employed within the proposed framework but their inability to identify patterns effectively makes them less attractive in our case. Moreover, recent analysis in [30, 31] pointed out that such techniques do not provide the levels of privacy that were previously thought to exist. Their experiments revealed that, in several cases, these techniques offer very little privacy. Traditionally, secure multiparty computation techniques have very poor scalability characteristics because of highly interactive matters of underlying protocols, which make them impractical for such evolving services. One important aspect in privacy research dealt with recently is the implications of social coalitions to attain better privacy. Work in [6, 7] suggests the importance of implicit groups between users in managing their privacy. Implicit groups are created within end-users to fulfil a specific goal that will provide

"anonymity". This categorization is personal to group members and not known to others. However, entities operating within a coalition may be wary of providing other coalition members with high fidelity data at all times for numerous possible reasons. Therefore, before providing others with this data, they may conceal it in order to reduce its accuracy. Our proposed collaborative privacy approach relies on a distributed topology of participants, where participants are organized into peer-groups and each peer-group contains a reliable peer to act as a trusted aggregator that is an entitled super-peer who will be responsible for anonymously sending the aggregated data of members within this peer-group to the social recommender service. In addition, after receiving the referrals list, the super-peer will be responsible for distributing this list back to its peer-group. Electing these super-peers is based on negotiation between participants and a trusted third party. This trusted third party is responsible for generating certificates for all participants, and managing these certificates. In addition, it is responsible for making assessments of those super-peers according to participants' reports and periodically updates the reputation of those super-peers.

The proposed collaborative privacy approach attains anonymity and privacy. The anonymity is achieved by either using an anonymity network like Tor or by dividing system users into a coalition of peer-groups. Each peer-group is to be treated as one entity by aggregating its members' data into one aggregated profile at the super-peer. This super-peer will then handle the interaction with the social recommender service. Individual participants might benefit from this anonymity to contact the recommender. If profiles cannot be identified and assuming that the initial user cannot be traced back, the system protects the privacy of the users even if profiles are sent in clear. However, participants' data privacy is achieved as each participant within the peer-group performs at least one stage in the concealment process based on his/her role in the peer-group. Traditional members perform a local concealment process before releasing their data to external

entities. Local concealment is a pre-processing step that is based on clustering the sensitive data. It then applies a concealment algorithm on the extracted partitions, so as to take into consideration the correlations and range of different data cells within sensitive data. Super-peers of every peer-group aggregate the data received from traditional members then execute a global concealment process on this aggregated data before releasing it to the service provider. This sort of two stage concealment process enforces anonymity for participants' identities and privacy for their data. Data privacy in the two stage concealment process was achieved by using a set of newly proposed stochastic techniques for concealing users' personal data within their released profiles. This is not a straightforward task since the two stage concealment process should make sure that the concealed data is still useful for the recommendation phase, which usually requires that changes in the users' personal data be as limited as possible. However, users' profiles are complicated and are an interrelated structure. Making small changes in it could cause an unexpected influence on the overall recommendation process. The proposed techniques combine approaches from the machine learning clustering analysis that consider knowledge representation in the domain of data privacy in order to preserve the aggregates in the dataset to maximize the usability of this data, with a view to accurately perform the desired recommendation process.

Furthermore, the effectiveness of the proposed collaborative privacy framework in solving feasible business applications was demonstrated on numerous scenarios associated with collaborative filtering based recommender services as an example for social recommender services. Collaborative filtering is a method executed in a social manner to extract automated recommendations based mostly upon the assumption that users who share similar preferences in the past will probably also share similar preferences in the future. It is a crucial task performed at online social service providers for recommending relevant items to their users. Online social service providers might run the recommender service as a

part of their network but they are required to buy, build, train, and maintain their recommender service infrastructure despite exponential costs. Moreover, in order to run this service well, providers need to recruit a highly specialized team to tune and handle ongoing problems that arise once the service runs. With the recent advance of cloud computing, several online social service providers opt for an outsourcing service model since it enables them to overcome their lack of computational power or expertise. They can plug in and subscribe to a third party service provider running the recommender service built on shared infrastructure via the Internet, where the user's data is outsourced to this recommender service to perform the desired processing. The recognition of the outscoring service model is steadily increasing because it simplifies deployment and reduces client acquisition costs. The multi-tenancy feature of those online services permits content providers to scale as quick and as much as needed without replacing costly infrastructure or adding IT staff.

Privacy is the main concern for these online social service providers since service providers might be situated abroad with totally different legal structures and data privacy laws. In practice, users have shown an increasing concern for sharing their private data, especially in the case of untrusted parties [4]. Therefore, the main challenge is to design an efficient privacy mechanism that shields against unauthorized access to user's personal data, while at the same time exposes a sufficient amount of information to the third party recommender service in order to receive useful recommendations. Among several existing approaches to recommender systems that pride themselves in providing accurate recommendations, only a few tackle the privacy issues and aim to manage the privacy risk of recommender systems as addressed by [32]. In addition, several of the existing recommender systems which are based on multi-party recommendation protocols, did not take into consideration the privacy issue. During this thesis, we have developed a non-cryptography based privacy enhancing

technologies for multi-party recommendation problems, whereas existing traditional cryptography based recommendation algorithms can be used.

Employing outsourcing for an online recommender service is a privacy hazard as both the data of content providers and users are fully under the control of a third party service provider. In this thesis, we intend to show that outsourcing for online recommender services can be applied efficiently through the proposed framework. We identify privacy risks associated with these services and propose remedies considering the following scenario. Current social recommender services providers have their business models centred on the availability of the users' personal data at their servers. Our scenario complies with this model whereby users' personal data is assumed to be stored on servers in a concealed form so as to minimize the privacy risks. However, the collaborative privacy approach facilitates sharing data between users in a peer-to-peer fashion within peer-groups.

It has been widely recognized [33] that peer-to-peer data sharing is a desired alternative for server based data sharing solutions for reasons pertaining to risks to user privacy. The collaborative privacy approach equipped with stochastic techniques ensures a privacy enhancing recommendation process such that user personal data never leaves his/her device in a raw form. For this purpose, a collection of purpose specific stochastic techniques were proposed to be executed before releasing the data outside of the users' devices. This approach eliminates the risk of possible privacy abuses as the sensitive data is only accessible to the owner but not to the other parties. The structure in data is destroyed as a consequence of applying our techniques. However, in order to facilitate performing the desired recommendation task, our techniques maintain some properties in this data which is required in the planned recommendation techniques. The aggregated profile at the super-peers is properly concealed before being shared with the recommender service. As a result, the recommender service provider holds a provisional snapshot of the users' personal data

in a concealed form. The challenge is to apply off-the-shelf recommendations techniques on data in this concealed form to obtain accurate recommendation results. The recommender service needs to collaborate with super-peers to update the users' personal data and to build an accurate recommendation model. Throughout the entire process, the recommender service neither learns about the actual data of individual users nor whether or not he/she is a member of any peer-group. Once the two stage concealment process is complete, recommendations are served to individuals based on the aggregate profile of the members of the same peer-group and results are shared between all members within each peer-group.

To summarize, the abstract goal of this thesis can be formally delineated as follows,

To develop a highly efficient collaborative privacy solution, which will aid in reconciling the need of data sharing for social recommender services with the increasing demand of privacy protection. The proposed solution can facilitate the solving of fine grained personalization problems in a wide choice of application domains on a cross system personalization. Providing a sensible distributed clustering algorithm over vertically partitioned datasets while taking into account privacy of such data is also a goal of this thesis.

1.3 Research Challenges

Despite the benefits of recommender services in various scenarios, new privacy threats exist while making use of them especially when combined on the social network side. Bringing various data together to support these services makes misuse easier, yet in the absence of adequate safeguards, the frequent use of these services can jeopardize the privacy and autonomy of participants. Privacy invasion occurs when individuals are unaware of "behind the scenes" use of their personal data. The simplest forms of privacy invasion by recommender service providers are unsolicited marketing, customer

segmentation, and scoring [34]. Data collected from participants is a valuable asset, and it can be sold when providers suffer bankruptcy.

When we started to design privacy enhancing technologies (PETs) for the proposed collaborative privacy approach, we discovered that designing PETs for social recommender services is different than designing PETs for any other kind of services. This is due to the inherent problems that exist within each service that limit reusing conventional versatile PETs across various services. Therefore, we needed to understand the major problems that exist in social recommender services that impose designing unique PETs to mitigate or avoid some of them. Good privacy enhancing technologies should be able to elaborate in any kind of social recommender services and produce accurate referrals. In this section, the challenges will be presented in designing PETs for data privacy in collaborative approaches.

Challenge 1: Attaining data privacy (C1)

The main concern for users while utilizing an external service is to preserve the privacy of their sensitive data during this process. Participants expect to define what data to release or share for the service and what data to hide. *A good PET should allow the participants to specify their own privacy requirements and control what to share over their data. Moreover, the released data should be somehow concealed (using tools like PETs, anonymity networks, and/or pseudonymous) in order to not to be linked to its original version. In addition, the proposed PET should attain privacy by combining efforts from both technical and legal domains.*

Challenge 2: Accuracy of results (C2)

When users utilized an external service, they expected to receive accurate results. In order to achieve this, *a good PET should seek to diminish the variance and bias of the data. Additionally, the good PET should provide parameters to control the level of concealing of*

released data, to implicitly inform the user about the expected accuracy he/she might get in return to this desired level.

Challenge 3: Diversity of users' profiles (C3)

With the exponential daily growth in the number of items offered by the majority of content providers, participants usually are exposed to a small proportion of items in relation to the total number of items. An effect of that is that participants' profiles become sparse which can cause difficulties in measuring similarities between participants and executing PETs properly. *A good PET should weight this problem and try to mitigate its effects on both of the concealment process and recommendations.*

Challenge 4: Interoperability (C4)

The social recommender service needs heavy processing which increases in proportion to the number of users and items available in the content provider. An effect of this is that the social recommender service will not be able to extract sensible model/referrals in a reasonable time. Moreover, this will hinder the ability of the recommender service providers to lease their services as much as possible to content providers. An alternative solution to this problem would be to utilize the social nature of these content providers and employ distributed aggregation protocols. However, this comes at the cost of the quality of generated model/referrals. *A good PET should be able to execute in either a centralized or distributed manner. Additionally, the good PET should be able to conceal accumulated data in scenarios where aggregation protocols are being applied.*

Challenge 5: Redundant Items (C5)

In numerous content providers, it is possible that different names might return to the same content. Unreasonable referrals can be generated if this problem exists. Moreover, synonyms can cause a severe privacy invasion, as an individual participant might set a rule to prevent the release of a certain item, but this item might have different names within the content provider. *A good PET should consider a way to mitigate the effect of the synonym problem on privacy invasion.*

Challenge 6: Global Information Sharing (C6)

New participants of the content provider cannot join a recommendation process without having a sensible profile. Current social recommender services force the new participant to rate a predefined set of items. Although this problem diminishes after a period of system usage, new participants will not be able to receive referrals during this period. *A good PET should consider methods to allow new participants to receive referrals based on querying other participants in order to reduce their waiting period and complete their profiles quickly.*

1.4 Research Scope of the Thesis

The complex issues related to data privacy in social recommender services cannot simply be addressed by deploying secure channels, restricting data collection, or keeping the privacy-sensitive data of the users encrypted on the server side. While these security measures eliminate a number of security threats, they are not sufficient to protect the sensitive data against misuse by the recommender service providers. There is no exact solution that resolves these privacy issues. However, an approximate solution could be sufficient, depending on the service, since the appropriate level of privacy can be interpreted in different contexts [32]. In the case of social recommender services, an appropriate balance between a need for privacy and accuracy can be found.

The general philosophy that we have taken in this research will be focused on principled solutions to protect the privacy of users in social recommender services. For this purpose, we propose to keep the personal sensitive data safe by means of the purpose specific two stage concealment process within a middleware framework. This two stage concealment process employs a set of stochastic techniques (PETs) derived from the machine learning clustering analysis to conceal the sensitive data while preserving the desired aspects in this data for the required recommendation technique. In addition, the middleware helps participants to form a coalition where they can interact with each other in a P2P fashion.

Participants within this coalition form a virtual topology to aggregate their data and then provide this aggregated data to the social recommendation service. The protection of privacy is attained using a set of stochastic PETs which destroy the structure of the data but, at the same time, maintain some properties in it which is required in the planned recommendation task. Two main threat models were employed to mimic attacks on the proposed two stage concealment process, mainly, the Graph Matching [126] and Re-identification [127] Attacks. Within the Graph Matching Attack [126], the untrusted social recommender service is trying to link the group profiles' data to certain users in the peer-group while within the Re-identification Attack [127], the untrusted social recommender service in collaboration with a malicious peer wants to filter out the existing collected items from a group profile based upon a portion of the originally released items which have been disclosed by a malicious member within a peer-group in order to discover if certain preferences were released by the victim's profile

The proposed middleware helps users to manage their personal privacy, empowering them with choice and prior consent so they can choose to share the appropriate information, with the preferred participants, in the desired recommendation request. The implementation of such an approach also confirmed that it is feasible to make use of and, at the same time, to protect the personal sensitive data of individuals, and to do so in an accurate way. The contributions of this thesis will be presented in Section 1.6.

1.5 Research Questions

The research presented in this introduction explores the new objectives of protecting the privacy of sensitive data while successfully coping with the current outsourcing model of social recommender services. The objectives of this thesis are represented by the following research questions:

Q1. How can we build a clustering model on data distributed between multiple sites bearing in mind the privacy issue? Furthermore, how can we measure the validity of this clustering model in practical scenarios? Challenges (C1 and C4)

Q2. What threat models can be utilized in third party social recommender services when users release their real profiles to earn accurate referrals. Furthermore, can OECD privacy principles be elicited for developing practical PETs for social recommender services? Challenges (C1 and C2)

Q3. Without the need to trust provider's declared policies and self-regulations, what framework can support protecting the users' privacy before their data is shared with social recommender services such that this framework maintains privacy and anonymity for participants. Furthermore, what practical scenarios can benefit from the whole architecture? Challenges (C1, C4, and C6)

Q.4 What framework can support privacy in collaborative platforms such that a recommender service can leverage the databases of different competing online database providers to provide better referrals without breaching the privacy of their users? Furthermore, what application can benefit from the whole architecture? Challenges (C1, C2, C3, C4, and C5)

Q5. How can privacy be enhanced in third party social recommender services by technical means with a reasonable trade-off between privacy protection and accuracy loss? Can these technical means transform the original data into a new one that conceals sensitive data while preserving the required patterns for an accurate recommendation task? *Furthermore, can we develop a non-cryptography based technical means for multi-party recommendation problems so that existing traditional cryptography based recommendation algorithms can be used?* Challenges (C2, C3, and C5)

1.6 Thesis Achievements & Contributions

The recent advances in mobile technology and communication infrastructures and at the same time the increasing need for these technologies for everyday tasks of individuals have expanded the amount of personal data that has been loaded to a new wide range of social recommender services. The main goal was to propose new innovative applications that shield the individual's privacy while utilizing users' personal data. Throughout the process of realizing the thesis goal, significant contributions in numerous application scenarios were identified, which demonstrate that personal data can be protected and can concurrently be utilized successfully within significant applications. The main contributions of this PhD thesis can be summarised as follows:

A1. **Clustering Algorithm:** We have developed an efficient distributed clustering algorithm (DLC) that bears in mind the privacy issue. The model building mechanism in the DLC is based on two mechanisms that only use the statistics about the data without the need to gather all the original data from each site. A privacy enhancing version of the DLC algorithm using cryptographic protocols for hospitals' collaborations was also presented. Thereafter, a private monitoring application for the analysis of ECG data within a personalized healthcare system was given. The DLC was utilized to gather statistics about ECG data. A clustering model was then built based on these statistics to enable a private analysis of ECG data in order to discover specific cardiovascular disease patterns.

A2. **Privacy Attack Model:** Due to the lack of standard threat models for data obfuscation techniques, we have investigated various threat models in the graph data and then have adopted two possible privacy threats within social recommender services, particularly in the area of third party recommender services for IPTV networks and sub-community discovery providers. Threats are against service users, in the context of releasing their profile data to obtain recommendations. Furthermore, a new set of machine learning

based stochastic techniques was proposed to mitigate the effect of these attacks. Finally, we evaluated Graph Matching [126] and Re-identification Attacks [127] on data concealed using our newly proposed stochastic techniques for various scenarios in order to demonstrate the stability of our proposed techniques against such attacks.

A3. **Collaborative Privacy Framework:** We have developed a collaborative privacy framework based on a two stage concealment process that involves novel algorithms and protocols. The related components within this framework were identified and the interaction between these components was described in detail in Section 3.3. The two stage concealment process uses a set of machine learning based stochastic techniques that conceal the data in a specific way so as to retain its statistical content while concealing all private information. As a result, privacy is achieved for both individual participants and peer-groups. The proposed framework was applied as a middleware which combines all of these techniques to make it possible to efficiently take advantage of this work. The proposed middleware enables participants to be organized in a distributed topology to achieve anonymity for participants with a relatively low accuracy loss. However, this topological formation prevents the service provider from creating a centralized database with raw personal data from each user and permits a decentralized execution of a two-stage concealment process on the users' personal data. In addition, this topological execution in the proposed middleware satisfies the requirements of high scalability and reduces the risk of privacy breaches. The proposed framework is vulnerable to malware/spyware that might infect the user's machine. In order to mitigate this problem, special considerations need to be added to the operating system of the user's machine in order to ensure strong safety and trustworthy guarantees to the middleware in the running memory and storage of the users' machine even in the presence of a malicious software (sandboxing, antivirus…etc.). Moreover, the proposed framework has some potential exposes while

interacting with other services within the content provider network. Specific solutions are required to attain privacy for each service at the content provider side. For example, when the user interacts with other third party services such as payment service at content provider side, the billing data can be used to reveal the user identity. But, solutions such as anonymous electronic cash, P2P money, and E-wallet systems can be easily integrated into the framework to prevent side-channel threats and attain full privacy. In particular, the E-wallet module can be added at user device as an agent within the framework in order to hold the electronic cash that is required for billing. In another example, the user interacts with other third party services such as content distribution service at the content provider side. The consumption history of each user can be used to reveal the users' preferences and identity if it is linked to the received recommendations. But, solutions such as secure content transmission protocols and P2P overlays within the peer-group can help to attain privacy for the user's consumption profile and can be easily integrated into the framework. Finally, the validity of the framework is demonstrated by the implementation and evaluation of the proposed solution within a set of important innovative applications including recommender service for IPTV content providers, data mash-up service for IPTV recommender service and community discovery and recommendation services for implicit social groups.

A4. **Data Mash-ups Service for IPTV:** We have developed a privacy aware data mash-ups service for IPTV recommender service as an enhanced application supporting the newly proposed collaborative privacy framework. This scenario introduced the concept of the 'K-Similar' perturbation technique that was developed with an aim to realize privacy by reducing the informational value of data items before revealing them. This enables better protection of privacy while still taking advantage of the offered services

(possibly with different levels of service quality). The whole scenario was evaluated to demonstrate the validity of our collaborative privacy approach.

A5. **Recommender Service for IPTV:** We have developed a privacy aware recommender service for IPTV content providers as an enhanced application supporting the newly proposed collaborative privacy framework. The proposed service takes into consideration the privacy of the users' personal data while generating recommendations. We proposed a set of PETs tailored for this scenario. The proposed collaborative framework utilizes different machine learning based stochastic techniques for the two stage concealment process. We also proposed a hybrid solution for protecting privacy in social recommender services based on secure multi-party recommendation. We introduce the usage of multi-level obfuscation and pseudonymization techniques to attain data privacy while preserving the desired aspects in the data for the planned analysis. The proposed multi-level obfuscation implements an access control mechanism based on trust heuristics, which enhances privacy by allowing different concealed copies of the same data to be released for various requests based on different trust levels. This adds an extra-layer of secrecy on the data and makes our secure multi-party recommendation very simple by having a low execution time for a big dataset. The whole scenario was evaluated to demonstrate the validity of our collaborative privacy approach.

A6. **Community Discovery & Recommendation Service:** We have developed a private community discovery & recommendation service within social groups as an enhanced application supporting the newly proposed collaborative privacy framework. The proposed service implements a model that captures the user's interests at his/her side, and then metric functions are used to calculate the similarity between users and communities in a distributed and private fashion. The proposed privacy mechanisms in this research conceal the user profile by means of the inspection of the contents of the

profile to extract certain aggregates. Users are connected together according to their similarity values, and then recommendations are provided to new users in order to join the discovered communities in their neighbourhood. However, we have also focused on applying our effective collaborative privacy approach on the users' profiles to support safe yet efficient data sharing for implicit discovery of communities and sub-communities in social groups. Additionally, in the scenario we considered the honest but curious attack model [49], where the main adversary is the community recommender service provider which the users send their preferences and requests to. Community recommender service is not aware of the identities of the members and their public/private profiles. In particular, the community recommender service does not have the knowledge of the distance function computed at super-peers side. The community recommender service only knows the sub-communities profiles which are represented using centroid and similarity values. For this attack model, the users within the peer-groups follow the rules of the protocols properly, without any deviations, and provide the correct inputs with the exception that they might keep a record of all intermediate values of computations. The whole scenario was evaluated to demonstrate the validity of our collaborative privacy approach. It worth to mention that another type of attacks involving malicious users [49] are out of scope of this thesis.

1.7 Document Organization

In this chapter, the background, motivation, and challenges of this research have been addressed resulting in a number of research questions. An overview of the relevant literature will be presented in Chapter 2. A summary of the research contribution will be presented in Chapter 3. Chapter 4 will then present the concluding remarks of the research work presented in this thesis and future work. Finally, the reprinted research papers containing the contributions of our research will be accessible in the appendices A to E.

Chapter 2

State of the Art

In this chapter, the state of the art that inspired the research conducted in this thesis will be presented.

The state of the art chapter is divided into three sections. Section 2.1 will present the basic concepts of cryptography utilized for the proposed PETs. Section 2.2 will present the state of the art of data clustering algorithms. Section 2.3 will provide a general overview of the state of the art of recommender systems and their taxonomy, and finally, Section 2.4 will describe the state of the art technological and legislative solutions for attaining data privacy.

2.1 Basic Concepts of Cryptography

In this section, the basic concepts of cryptography will be presented, which include (Section 2.1.1) the main idea of cryptography, (Section 2.1.2) homomorphic encryption, (Section 2.1.3) cryptographic hash algorithms, (Section 2.1.4) RSA cryptosystem, (Section 2.1.5) ElGamal cryptosystem, (Section 2.1.6) Paillier cryptosystem, (Section 2.1.7) public key certificate, (Section 2.1.8) transport layer security, as well as (Section 2.1.9) secure multi-party computation. Since the scope of this thesis focuses on the non-cryptography based PETs for preserving privacy, we only consider this part for multi-party recommendation problems so that traditional cryptographic based recommendation algorithms can be used with the proposed PETs efficiently, in order to add an extra layer of privacy for these algorithms without utilizing bigger key sizes.

2.1.1 Main Idea of Cryptography

The main theme of cryptography is to address the requirement of sending a message from a sender to a recipient with security in order to tackle the risks of it being read by others [35]. The initial message is the cleartext. The process of hiding data in cleartext is called encryption. The encrypted message is the ciphertext, and the process of extracting data from ciphertext is called decryption. The whole process is presented in Figure (2.1)

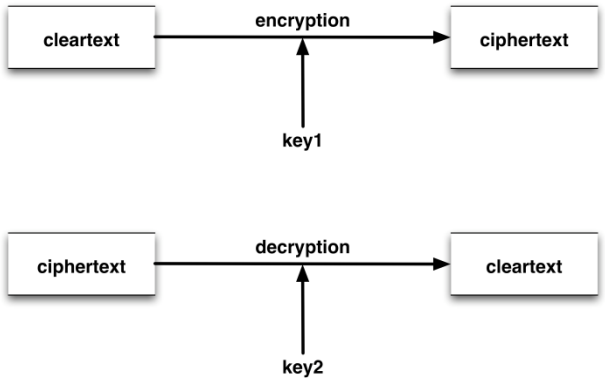


Figure 2.1: Encryption and decryption

The modern encryption system utilizes a key (or keys) during the procedures of encryption and decryptions. This key can be any of a large number of values, where the range of these possible values is called keyspace. The whole security of encryption algorithms relies on the special key (or keys) and not within the details of the algorithm. Some algorithms use completely different keys in the encryption and decryption, $key1 \neq key2$ as shown in Figure 2.1. These types of algorithms are called asymmetric algorithms or public-key algorithms. However, there are other types of algorithms where the same key is used in encryption and decryption which are called symmetric algorithms (also called secret key encryption algorithms), $key1 = key2$ as shown in Figure 2.1.

2.1.2 Homomorphic Encryption

Homomorphic encryption is a kind of encryption scheme which permits specific types of computations to be carried out on the ciphertext that will produce a new ciphertext. Once

decrypting this new ciphertext, it will produce an output plaintext corresponding to a desired operation on the input plaintext [36]. This property can have both positive and negative effects on the encryption scheme. The vulnerability of homomorphic encryption schemes to malicious attacks makes them unsuitable for secure data transmission. However, these schemes are often utilized to create widespread secure multi-party computation protocols.

Homomorphic encryption can be divided into two main classes. The first are the partially homomorphic cryptosystems that support only one operation either addition or multiplication operations on encrypted data and the second are the fully homomorphic cryptosystems [37] that support both addition and multiplication operations on the same encrypted data. In this thesis, the focus will be on the partially homomorphic cryptosystems, which are more practical and efficient. Moreover, so far fully homomorphic cryptosystems are not efficient enough to be employed in practical applications [38].

2.1.3 Cryptographic Hash Algorithms

A hash function is a function that receives an input data of arbitrary length and turns that into one way data which is hard to invert to return in the original form [39]. These functions can be regarded as compression functions which produce a hash variable (digest) with n length for any given input of m length as shown in Figure 2.2. Cryptographic hash functions are called secure hash algorithms (SHA). Until now, the SHA family consists of four members (SHA-0, SHA-1, SHA-2, and SHA-3).

Figure 2.2: Encryption and decryption

A recent family of hashing functions is the locality sensitive hashing (LSH). The idea of locality sensitive hashing was first introduced in [40]. The basic concept of LSH is to hash all the items such that similar items are hashed to the same bin with high probability. Intuitively, the hashing scheme projects all the items onto a random line and then divides this random line into multiple equal length segments to generate bins. Therefore, items that are closer will fall into the same bin with high probability, as shown in Figure 2.3.

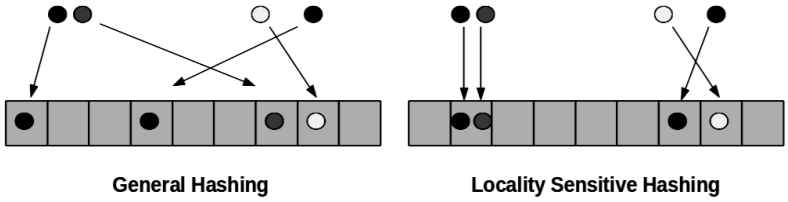


Figure 2.3: The difference between general hashing and LSH.

2.1.4 RSA Cryptosystem

A popular cryptographic system is coined in [41]. The RSA is a public key cryptosystem that has been until now computationally infeasible to crack. However, within the near future if quantum computers are constructed, this may be changed. The RSA algorithm consists of three main steps: the key generation, the encryption algorithm and the decryption algorithm.

Key Generation: The beginning procedure which is invoked to create key pairs (public and private).

1. Calculate $n = p.q$, where p and q are two distinct prime integers.
2. Chose a random number d such that $gcd(d, \lambda) = 1, \lambda = (p-1)(q-1)$
3. Determine e such that $(e.d)\ mod(p-1)(q-1) = 1$, so e is the inverse of d : $d^{-1}\ mod\ (p-1)(q-1)$.
4. The public key is (e, n) and the private key is (d, n)

Encryption: the encryption of message m using the public key(e, n): $\mathrm{E}_k(m) = m^e mod\ n$

Decryption: the decryption of message $c = E_k(m)$ using the private key(d, n): $D_k(c) = c^d$

The RSA algorithm fulfils the multiplicative homomorphic encryption property which implies that the multiplications of encrypted values correspond to the product of decrypted ones. Concretely, given the encryption of plaintexts $x_1, x_2 \in \mathbb{Z}_n$

$$E(x_1).E(x_2) = x_1^e x_2^e \ mod\ n = (x_1 x_2)^e mod\ n = E(x_1.x_2) \qquad (2.1)$$

In order to crack the RSA algorithm, the adversary needs to know the private key. We already know the number n from the public key (e, n). The task now is to find two prime numbers p and q which have the product n. It is practically impossible to factor a number into a product of two primes (factorization), particularly when the number has many digits. As a result, the primes p and q should be large enough so that the fastest factorization algorithm requires time longer than in which that data must be secured.

2.1.5 ElGamal Cryptosystem

A probabilistic asymmetric public key encryption algorithm is presented in [42] and is based on the concept of a discrete logarithm problem. ElGamal encryption employs randomness in the encryption algorithm such that when encrypting identical data several times, it will yield totally different ciphertexts. The ElGamal algorithm consists of three main steps: the key generation, the encryption, and the decryption.

Key Generation: The starting procedure which is invoked to create key pairs (public and private).

1. Choose a random large prime p and prime generator g from $Z_p^* = \{1,2, \dots, p-1\}$
2. Select a random number a as a private key from the interval $1 \leq a \leq p-1$
3. Calculate $y = g^a \bmod p$, then the public key will be (p, g, y)

Encryption: Encrypting a message m in the set $\{1,2, \dots, p-1\}$ using the public key (p, g, y) is done as follows, first by selecting a random number r in the set $\{1,2, \dots, p-1\}$

then calculate $\gamma = g^r mod\ p$ and $\delta = my^r mod\ p$, therefore the ciphertext is $E_k(m,r) = (\gamma, \delta)$.

Decryption: Decrypting the message $E_k(m,r)$ using the private key a is done as follows: First calculate the initial message $(\gamma^{-a})\delta\ mode\ p$, therefore the plaintext $D_k(\gamma,\delta) = m\ mod\ p$.

The ElGamal algorithm fulfils the multiplicative homomorphic encryption property which means that the multiplication of encrypted values corresponds to the product of decrypted ones. Concretely, given the encryption of plaintexts, $x_1, x_2 \in \mathbb{Z}_n$

$$E(x_1).E(x_2) = (g^{r1}, x_1.y^{r1})(g^{r2}, x_2.y^{r2}) = g^{r1+r2}, (x_1.x_2)y^{r1+r2} = E(x_1.x_2) \quad (2.2)$$

In order to crack the ElGamal algorithm, the adversary needs to recover the private key a from the public key(p, g, y). Therefore, the adversary needs to solve the discrete logarithm problem with base g. It is computationally infeasible to derive such significant information regarding the plaintext given only the cipher-text and public key. The size of the prime number p should be large enough in order to ensure the security of the data. However, this has two disadvantages including the increase of encryption time and the expansion of ciphertext.

2.1.6 Paillier Cryptosystem

A probabilistic asymmetric algorithm invented and presented in [43] for public key cryptography. It is based on the $n-th$ residue classes problem. The Paillier algorithm consists of three main steps: the key generation, the encryption, and the decryption.

Key Generation: The starting procedure which is invoked to create key pairs (public and private).

1. Choose two large prime numbers p and q such that $GCD\big(pq, (p-1)(q-1)\big) = 1$ and a random integer $g \in \mathbb{Z}^*_{n^2}$

2. Calculate $n = pq$ and $\lambda = LCM(p-1, q-1)$. To confirm n divides the order of g, we check the existence of the following modular multiplicative inverse $\mu = \left(L\left(g^{\lambda} \bmod n^2\right)\right)^{-1} \bmod n$, where function $L(u) = \frac{u-1}{n}$

3. The public key is (n, g) and the private key is (λ, μ)

Encryption: The encryption of message $m \in \mathbb{Z}_n$ using the public key (n, g) is done by selecting random number $r \in \mathbb{Z}_n^*$ then $E_k(m, r) = g^m . r^n \bmod n^2$.

Decryption: The decryption of message $c = E_k(m, r)$ using the private key (λ, μ) is $D_k(c) = L\left(c^{\lambda} \bmod n^2\right) . \mu \bmod n$.

The Paillier algorithm fulfils the additive homomorphic encryption property which means that multiplications of encrypted values correspond to the sum of decrypted ones. Concretely, given the encryption of plaintexts $x_1, x_2 \in \mathbb{Z}_n$

$$E(x_1).E(x_2) = (g^{x1}.r_1^n)(g^{x2}.r_2^n) = g^{[x1+x2 \bmod n]}.(r_1.r_2)^n \bmod n^2$$

$$= E([x1 + x2 \bmod n]) \qquad (2.3)$$

The Paillier algorithm is secure against the chosen plaintext attacks (IND-CPA) [44]. The adversary's ability to distinguish the challenge ciphertext amounts to the ability to decide composite residuosity. However, due to the homomorphic properties the algorithm is malleable. Thus, it is not secure against adaptive chosen ciphertext attacks (IND-CCA2). An improved scheme which is secure against IND-CCA2 was proposed in [45].

2.1.7 Public Key Certificate

In cryptography, a public key certificate or identity certificate [46] is a certificate that uses a digital signature to bind together a public key with the identity regarding an entity. This certificate can be checked whether a public key belongs to this entity or not. The certificate authority is the issuer of that signature to attest that identity and the public key belong together. The most common certification standard is the ITU-T X.509. The certificate can

be revoked if the relationship between the entity information and the public key is found to be inaccurate. This can be done either by comparing this certificate with the certificate revocation list or querying the certificate authority (CA) using the online certificate status protocol.

2.1.8 Transport Layer Security

Transport Layer Security (TLS) [47] is a cryptographic protocol that provides communication security over the Internet. Several versions of this protocol have been deployed to be utilized within different applications. TLS protocol provides isolated communication over the Internet as a way to prevent eavesdropping and tampering. In a typical model, each client should indicate whether he/she wants to setup a TLS connection or not by specifying the port number for TLS connections (for example, port 443 for https) or specifying regular ports where the server switches the connection to TLS using specific mechanisms (for example STARTTLS).

2.1.9 Secure Multi-Party Computation

The secure multi-party computation (SMPC) is a problem in cryptography that was first introduced by Yao in [48] using the millionaire's problem as an example to describe SMPC. Yao proposed a solution that permits both millionaires to satisfy their curiosity in finding out who is richer without revealing the precise amount of their wealth.

The millionaire's problem and its solution gave way to a generalization to multiparty protocols. In an MPC, a given number of parties $P_1, P_2, \ldots, P_N$ each have private data respectively $D_1, D_2, \ldots, D_N$. The parties want to compute the value of a public function $\mathcal{F}$ of N variables. An MPC protocol is secure if no party can learn more from the description of the public function and the result of the global function. More precisely, the protocol is secure if the following conditions hold [49]:

1. Completeness: If all parties honestly follow the protocol then they obtain an output as correct as the computation of $\mathcal{F}$ on $D_1, D_2, \ldots, D_N$.
2. Input/output privacy: Any party behaving dishonestly during the protocol does not gain any information about the private inputs /outputs of the other parties.

The dishonest behaviour in security protocols models possible real world attacks from adversaries. The dishonest behaviours can be classified as follows [49]:

1. Honest-but-curious model: The dishonest party must follow the protocol but can arbitrarily analyse the outputs of the protocol offline in order to infer some additional information.
2. Malicious model: The dishonest party can arbitrarily deviate from the protocol and corrupt it.
3. Computationally bounded/unbounded model: The computational setting and running time of the dishonest party is bounded by a polynomial. However, the unconditional setting puts no restriction on the running time of the dishonest party (i.e., it may be an exponential).

Proving that a protocol achieves input/output privacy is done by using a real world paradigm, where a trusted third party (TTP) exists that computes function $\mathcal{F}$. Parties do not communicate with one another but they send their inputs to the TTP and receive the outputs. In order to prove input/output privacy, it is required to show whatever a dishonest party [49] can infer about inputs/outputs of the honest parties by exploiting the protocol execution.

2.2 Data Clustering Algorithms

This section will present an overview of the state of the art in data clustering algorithms. Section 2.2.1 will describe why the clustering analysis was employed in this research and

Section 2.2.2 will illustrate the taxonomy of clustering methods. Finally, distributed clustering will be presented in Section 2.2.3.

2.2.1 Why Clustering Analysis

In this thesis, machine learning clustering methods were used as the basic building block for pattern preserving in the proposed PETs. Before applying PETs on the dataset, the dataset is pre-processed with the clustering method proposed in Article II [50] in order to extract essential patterns for recommendation techniques and then after the proposed PETs are adjusted to keep these patterns analogous to the ones in the original data. Utilizing the proposed clustering method enables the off-the-shelf recommendations techniques to extract these useful patterns directly from the concealed dataset without the need to modify these techniques to work with concealed data.

Clustering is an ideal pattern preserving method since it can extract regularities from the dataset. These regularities can be realized in the form of putting sets of similar objects into clusters, where each cluster has a representative or exemplar to it. An ideal clustering method seeks to attain two objectives: (1) minimizes the inter-cluster similarity and (2) maximizes the intra-cluster similarity. Choosing clustering analysis for this task has been done for three reasons:

1. An ideal clustering method has a predictive capability, where similarities between a new object and existing clusters can be utilized to identify the type of object based on which cluster it belongs to.
2. An ideal clustering method can reduce computation costs, since the whole cluster's objects can be represented by an exemplar, which is enough to describe the whole cluster structure.

3. An ideal clustering method extracts groups of objects that deserve attention while cleaning the data from outliers and errors.

2.2.2 Taxonomy of Clustering Methods

In this section, the four main categorizes of clustering methods will be presented [51], which includes: (Section 2.2.2.1) partitioning clustering, (Section 2.2.2.2) hierarchical clustering, (Section 2.2.2.3) density based clustering, (Section 2.2.2.4) grid based clustering, as well as (Section 2.2.2.5) model based clustering.

2.2.2.1 Partitioning Clustering

Partitioning clustering tries to transform a set of N objects into a set of k clusters such that this transformation optimizes a certain objective function. The methods in this type can be further classified as either static partitioning clustering or dynamic partitioning clustering [52]. Static partitioning is a method performed on the dataset in advance in order to extract a clustering model, and then the resulting clusters remain fixed during the whole system run. However, the dynamic partitioning is a method that extracts the clustering model during the system run. Different algorithms can be classified under the partitioning clustering type. The *K*-mean algorithm [28] is where each cluster is represented by its centre of gravity. The other variations of the *K*-means algorithm are the *K*-medoid algorithm [28], where each cluster is represented by an object near its centre, and the *K*-median algorithm, where each cluster is represented by its median object. Algorithms like PAM [53], CLARA [53], and CLARANS [54] are extensions to the *K*-medoid algorithm. The quality of the model generated by these algorithms relies on the selected objective function that is employed to measure the similarities.

2.2.2.2 Hierarchical Clustering

The hierarchical clustering attempts to transform a set of N objects into tree decomposition. The clusters in this case are a nested hierarchical sequence, where a single global cluster is at the top of the tree and one object cluster is at its bottom. The visualization of this decomposition is done by a dendrogram. Hierarchical clustering [55] recursively splits the set of N objects into small-size subsets until each generated subset contains only one object. The methods in this type can be further classified as either agglomerative clustering or divisive clustering. In the agglomerative methods, the hierarchical sequence is created from the bottom leaves up to the top root. However, with divisive methods, the hierarchical sequence is created from the top root down to the bottom leaves.

Agglomerative clustering begins with allocating each object in the dataset to a separate cluster, then it starts to recursively merge similar cluster pairs until attaining the required number of clusters. Divisive clustering begins with allocating all of the objects in the dataset to one cluster, and then it starts recursively splitting the data in each leaf cluster.

Different algorithms can be classified under the hierarchical clustering type. BIRCH [56] is a single scan algorithm that uses a CF-tree structure to split the dataset incrementally. CURE [56] is an agglomerative algorithm that uses random sampling and a fixed number of objects to represent a cluster. Inter-cluster is calculated using the closest pair of these objects that belongs to other clusters. These algorithms produce arbitrary shape clusters but they are sensitive to the order of data.

2.2.2.3 Density Based Clustering

Density based clustering identifies clusters by utilizing a density based function that reflects the spatial distribution of the objects within a region in the dataset. The notion of clusters is defined as density connected objects that are maximal in relation to reachability distance. The density of objects within the cluster is higher than outside it.

Different algorithms can be classified under the density based clustering type. DBSCAN [57] is an algorithm that relies on the assumption that s cluster is a region where the density of objects inside it is higher than the density of objects outside it. It uses two parameters that represent the radius and the minimum number of objects inside the cluster. OPTICS [58] is an interactive algorithm that creates an ordering of the data based upon its density based clustering structure. OPTICS is an extended version of DBSCAN, but instead of assigning objects to clusters, it stores them in the order that they were processed.

2.2.2.4 Grid Based Clustering

Grid based clustering [56] starts by first quantizing the cluster space into a finite number of grids and then executes the desired process on the quantized space. These algorithms have a fast processing capability, depending on the number of cells in each quantized dimension in the cluster space and the remaining independent objects. The cells containing a number of objects more than a certain parameter are to be treated as dense cells, then these dense cells are connected to each other to build a cluster.

Different algorithms can be classified under the grid based clustering type. STING [59] is an algorithm that divides the data space into several levels of cells in order to form a tree. The cells in the higher level consist of cells in the lower level. This algorithm relies heavily on the granularity of the lowest level cells. WaveCluster [60] is an algorithm that utilizes wavelet transform to transform the original feature space into other space where natural clusters within the data can be identified. As a result, this algorithm does not require the number of clusters as an input.

2.2.2.5 Model Based Clustering

Model based clustering [56] treats the clustering analysis as a supervised learning from incomplete data and employs a model derived from the statistical distribution of the dataset. Different algorithms can be classified under the model based clustering type. The Auto-

Class algorithm [61] uses a Bayesian method to extract clusters from those datasets. First, it assumes that the dataset is generated according to the probability distributions like Gaussian, Poisson, and Bernoulli. The algorithm assumes that there is an unobserved variable within the data that reflects the cluster membership for every case. It starts with a random initialization of its parameters then incrementally adjusts them to find their maximum like-hood estimates. The Auto-Class algorithm is an example of one that extracts optimal clusters based on the prior probabilities. A self-organizing map neural network (SOM NN) [56] is another model based clustering that employs a two layer neural network, where each neuron of the SOM NN is represented by n −dimensional input vector. During the training, the neurons are treated as cluster centres map units that try to form bigger clusters iteratively. The algorithm is robust and can extract clustering with arbitrary shapes.

2.2.3 Distributed Clustering

Several techniques have been utilized in the literature in order to attain distributed clustering. *K*-means algorithm has been broadly utilized to achieve distributed clustering by various scientists. A distributed clustering based on *K*-means was proposed in [62]. Their technique is equipped with a synchronization mechanism for updating the clustering model. The number of rounds in the synchronization mechanism is related to the iterations in the *K*-means algorithm. Another variation of this algorithm was presented in [63] to cluster data streams in P2P environment. Their technique is equipped with a sampling mechanism to reduce the synchronization phase. Based on these two techniques, numerous variations have been presented. The research work in [64] proposed an enhanced version of the *K*-means algorithm in P2P networks with the ability to run in an asynchronous manner. Authors in [65] presented a parallel version of the *K*-means algorithm running on a homogenously distributed dataset.

2.3 Recommender Systems

With the amount of information available on the internet about contents and products, it has been necessary to employ software systems which offer similar or better recommendations than humans. These systems later became known as automated recommender systems, which have been used to aid users in their decision-making regarding finding relevant products or items for them. The application area of recommender systems is huge and commercially extensive [66], providing solutions in diverse domains such as financial products, real estate, movies, books, music, news, friends, websites, and even groups to join or socialize with. An early recommender system for an e-commerce application can be found here [26]. Several challenges currently exist in the design of recommender systems. For instance, some users' preferences can be irrational or have incomplete user profiles, which can lead to generating inaccurate or overdue recommendations. Obtaining useful recommendations require the construction of extremely detailed user profiles that contain a great amount of personal information. This information will be stored in a centralized database of a recommender service provider that users have to trust. Many people won't like that their personal data will be stored in a remote site without protection. In addition, the recommendation is a very personal process that can reveal considerable personal information. The final recommendations depend on the users' profiles, so if the users provide inaccurate profiles then the recommendations are going to be less accurate. This thesis is centred on designing a private social recommender service such that the participants of the systems can preserve their privacy against an uncontrolled recommender service provider and other users collecting their profiles. The proposed approach does not affect the accuracy of the recommendations. We employed the formation of coalitions between users before starting the recommendation process. This is considered to be a more secure approach than interacting directly with a centralized social recommender service because coalitions prevent the creation of a single and complete database with all

information and avoid the usual centralized security attacks. Moreover, coalitions can be scaled to serve millions of users faster and cheaper than direct interaction with a centralized service. In the next sections, a definition will be given to a recommender system in Section 2.3.1. Section 2.3.2 will illustrate the new trend of utilizing recommender systems as services. Finally, Section 2.3.3 will describe the taxonomy of recommender systems.

2.3.1 What is a Recommender System?

For a long time, crawlers have been utilized for identifying, collecting, and parsing as many websites as possible. These crawlers insert this gathered data into a huge database. Users can send a list of keywords to this database in order to retrieve a set of webpages containing these words. This model works well as long as users know in advance a list of specific and well-defined words for the webpages they are looking for. Preparing this list of well-defined words is in some cases a difficult issue to realize. However, recommender systems aid the users in finding interesting and previously unknown webpages.

With the number of decisions the users have to confront on an everyday basis [67], gathering all the information to make a well-grounded decision is a very time consuming process. Recommender systems appeared to create a model of users that captures their preferences in order to assist them in quickly making the proper decision and saving time and money. As a result, recommender systems can be referred to as experts and they have been used in crucial fields like health services, financial investments, and e-learning. It is believed that recommender systems can substitute for experts, not only because employing automatic recommender systems is cheaper than hiring an expert [66], but also because the referrals that are generated can outperform the advice of an expert.

To summarize, different definitions for recommender systems can be found in the literature [67]. However, in this thesis, we will consider the definition of recommender systems as an automatic service that, given a user u and a set of items S some of them are unknown by u.

The recommender service generates a personalized subset including the most interesting items of S for the user u. From this definition, we can infer that recommender services are automatic systems that generate personalized results that the user was not aware of. Our view in this thesis takes into consideration that recommender services should also take care of the privacy for participants during the recommendation's process. Privacy is a necessity for a recommender service, since the recommendation process requires a detailed view or profile of each user. These profiles include sensitive information which captures the personal description of a particular user. Thus, privacy concerned users may be afraid of declaring their real profiles or release fake profiles, which in both cases impact the accuracy of generated recommendations.

2.3.2 Software-as-a-Service Recommender System

A recommender service provider is a new business model [68] which realizes a third party company offering a recommender service to a set of clients of registered content providers over a network. Examples of such outsourced recommender services include: Easyrec©, Directedge©, Starthq©, Barilliance©, Liftsuggest©, Smartfocus©, Sugestio©, and Myrrix©. These services host users' data from various content providers then employ various techniques in flexible and transparent configurations in order to extract referrals that are delivered through APIs. These services can be scaled to serve multiple content providers, such that, this large provider base leads to a cost reduction in service leasing in contrast to when in-house recommender systems were deployed and operated by the content providers themselves.

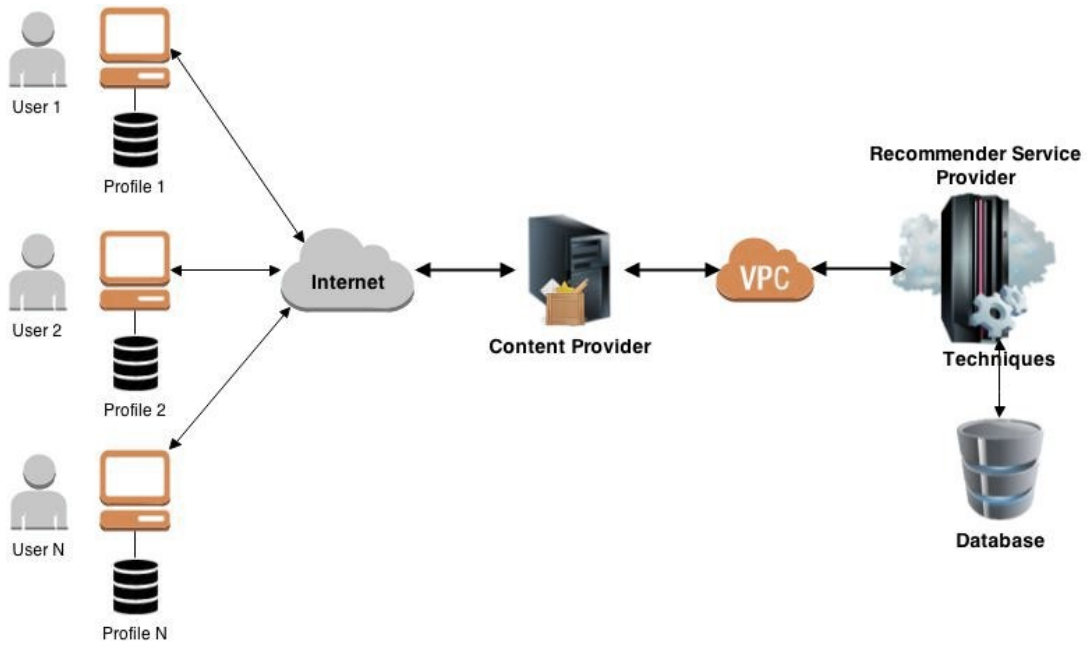


Figure 2.4: Architecture of recommender service provider

The architecture for a recommender service provider can be described as in Figure 2.4. The recommender service allows content providers to utilize its advanced computing resources, techniques, and expertise for generating referrals based upon data collected from users in their distribution networks. Content providers act as a mediator for accumulating users' profiles and delivering the recommended contents to their subscribers.

Generally, the collected profiles contain sensitive information, which raise privacy concerns for the users [69]. The main privacy threats of such a model are:

1. How can users make sure that the recommender service provider is trusted enough to handle their profiles' data in the same way that they announced?
2. How can users make sure that the recommender service provider prohibits unauthorized parties from accessing their data?
3. How can users make sure that the recommender service provider did not modify their data accidentally or intentionally?
4. How can users make sure that the recommendation process did not reveal sensitive information about them?

A reliable behaviour from the recommender service provider can be imposed by a contract, but due to various cases, the legislation alone might not be enough. The rapid developments in technology, differences between privacy laws, complex breaches in the infrastructure of the recommender service provider, and finally, the difficulty in detection and prevention of violations in the outsourced data, all limit the feasibility of any legislative efforts. The goal of this thesis is to propose a PET to ensure the privacy of the data outsourced to a recommender service as an input while allowing the extraction of accurate referrals from this data. The proposed PET conceals the outsourced data in a way which enables the recommender service to execute its desired recommendation technique on the concealed data yielding accurate referrals compared with the ones extracted from the raw data.

2.3.2.1 Steps of The Recommendation Process within The Recommender Service Provider

As we illustrated before, content providers might utilize a third party recommender as a service in order to provide referrals for their clients [66]. In this section, we formalize a general description of the steps that a recommender service utilizes for providing referrals. The actual systems may reduce some of these steps while others can perform complex steps within them:

- The content providers insert the profiles of items and descriptions about these items into the local database of the recommender service. These profiles contain the main characteristics of every item. This is an essential step since it allows the recommender service to identify the items that are going to be offered to the users.
- A profile is created and assigned to each user registered at the content provider. This profile could be controlled by the user or content provider. These profiles capture the preferences, ratings, and personal information regarding the system's users. Therefore, the information contained in these profiles is highly sensitive which imposes a

responsibility on the content providers to protect it in order to increase the trustworthiness of their services.

- Users send a request to the recommender service that includes their preferences. The complexity of this process varies with different recommender types.
- The recommendation algorithms/techniques are utilized in order to respond to the request, whereas recommenders search their internal database to select items that are appropriate to answer this request. The recommender service returns a set of identifiers for items that might be interesting to the users. These identifiers are linked to items offered by their content providers.
- During the final phase, users access the recommended items. These generated referrals may be useful to enhance the service and future recommendations, since accessing those items does imply that the recommendation was correct. However, from the security point of view, this is a security leakage whereas an attacker may learn of the user's profile by means of inspecting items that the user has utilized.

2.3.3 Taxonomy of Recommender Systems

Previous work in the literature created categories of recommender systems to classify all of them [66],[67],[68]. These categories are useful to understand the current trends in the recommender systems field. However, the practical recommender systems often share elements of several categories. In this section, a review will be made of the classification of the recommender systems based on three axes:

- Data source: The recommender system might use past behaviour information of the user as the main source to provide recommendations. It could also analyse items' profiles to decide which items might be interesting to the user. Section 2.3.3.1 will give an overview on this axis.

- Network structure: The recommender system can be centralized, decentralized, or hybrid in a social manner. Section 2.3.3.1 will outline this axis.
- Security: The recommender system might protect the privacy of the users or not throughout the recommendation process. Section 2.3.3.3 will describe this axis in detail.

2.3.3.1 According to the Data Source

According to the data source used to generate referrals, the recommender systems can be divided into:

- Collaborative or social filtering — where referrals are based on data collected from a network of users (social network).
- Content based filtering — where referrals are generated by direct inspection of items' profiles.
- Knowledge based recommendations — where referrals are performed by an expert system that inspects users' profiles.
- Demographic based recommendations — where referrals are performed based on demographic information within users' profiles.
- Hybrid algorithm — most realistic as referrals are generated based on a hybrid of the previous approaches.

Collaborative or social filtering is one of the most successful types of recommender systems where users help each other to detect interesting items. Recommendations based on collaborative filtering employ the ratings that other users made of the available items [66]. Therefore, it does not depend on the items' profiles. The collaborative filtering algorithms aim to guess the rating that a specific user will assign to an item by means of aggregating the ratings provided by other users, weighted with the similarity between users. Recommender systems of this type are based on these assumptions:

- Any two users that have common preferences in the past will have common preferences in the future. As a result, ratings that the users give for items in their profiles are useful to decide similarities between users and to decide whether or not a given item is interesting for each user.
- Only ratings are necessary for this type of recommender systems. Items' profiles are considered redundant and the recommender does not use these data in their recommendations.

The process of collecting ratings for items is hard and slow. It can be done using explicit or implicit methods. Explicit ratings collection is done by asking the user to do one or more from the following: rate an item on a sliding scale, rank items from highest favourite to least favourite, or select a set of items that he/she likes from a list. Examples of implicit ratings collection include: inspect the items that the user interacts with, analysing interaction between the user and items, or analysing the user's social network to discover similar users.

In a social network, a particular neighbourhood between the users can be calculated using the Pearson's correlation between these users in order to discover users with the same tastes or preferences and then assign them to the same neighbourhood [66]. Thereafter, the predication function on neighbourhood ratings is employed to obtain a rating of a specific user for an item that he/she had never experienced or consumed before.

$$sim(a,b) = \frac{\sum_{p\in P}(r_{a,p}-\overline{r_a})(r_{b,p}-\overline{r_b})}{\sqrt{\sum_{p\in P}(r_{a,p}-\overline{r_a})^2}\sqrt{\sum_{p\in P}(r_{b,p}-\overline{r_b})^2}} \qquad (2.4)$$

$$Pre(a,p) = \overline{r_a} + \frac{\sum_{b\in N} sim(a,b)(r_{b,p}-\overline{r_b})}{\sum_{b\in N} sim(a,b)} \qquad (2.5)$$

$sim(a,b)$ is the similarity of two users, a, b, $\overline{r_a}$ is the average rating of user a, and $r_{a,p}$ is the rating of user a of item p. $Pre(a,p)$ is the predicated rating for user a on item p. Another possibility is to provide a recommendation based on this neighbourhood by means of collecting the preferences of top-*N* neighbours of a specific user and then extract new

items from these preferences and offer them for the user. The approach of creating a neighbourhood has an advantage of reducing processing time by evaluating only items in that neighbourhood which is usually enough. Paradigmatic examples for systems that use collaborative or social filtering include: Amazon©, FilmAffinity©, StumbleUpon©, MovieLens©, Strands© and AggregateKnowledge©.

Content based recommender systems analyse the items' profiles to provide referrals to the users. It is strongly related to the information retrieval and filtering fields, where items are categorized in adjacent sets of interesting and not interesting according to their profiles [70]. It is based on the following ideas:

- Content based recommender systems classify items into two sets, interesting items and not interesting items for all users. This classification is based on items' profiles. The ratings that users give to some items are not relevant.
- Content based recommender systems takes as an input the item's profiles and optionally the user's profiles to measure the suitability of a recommendation to a specific user. As a consequence, this recommender can state the reasons for a specific recommendation.
- Content based recommender systems impose some internal structure to the items' profiles or the existence of specific metadata about these items. This information aids the recommender in making a decision about their relevancy.

The content based recommender system models the item using a profile, then it utilizes techniques such as Bayesian classifiers, neural networks, and cluster analysis [71] to classify these profiles. Examples of such systems that use content based recommender systems include: PANDORA©, Internet Movie Database©, and Webwatcher [72]. However, due to the limitations of this type of recommender system, currently social filtering techniques seem to be preferred by both the commercial and academic services.

Knowledge based recommender systems are closely related to the expert systems. The recommendations are generated based on inferences about the user's needs

and interests. Here, the knowledge is about the association of particular characteristics of the items to the current interests of a particular user [66]. These recommender systems are used when the user profile is not complete or the list of items that fulfil a set of characteristics is not enough. This type of recommender system depends on the following:

- A survey to identify the interesting items must be taken during each interaction with the recommender in order to facilitate the classification of items in a very specific way for each user.
- The use of the survey is an essential step since the system cannot trust the past history of the user to determine his/her interests. Moreover, this survey aids in capturing the relative importance of characteristics.

The knowledge based recommender system needs a complex interaction between the user and the recommender. Examples for such systems that use knowledge based recommender systems include: StyleHop©, ModCloth [73], Yelp© [74], and TrustPilot [75].

Demographic based recommender systems use demographic information [76] in order to identify the types of users that like similar items and require data on the individual users of the system. The key functionality of the demographic based recommender system is creating categories of users having similar demographic characteristics. It then tracks the aggregate preferences of users within each of these categories. Recommendations for a new user are made by finding which category he/she belongs in order to apply the aggregate preferences of previous users in that category. Clustering analysis techniques [77] have been employed for that role in the demographic based recommender system, where clustering is used to generate user categories mentioned above by considering the demographic characteristics of the set of all previous users. The aggregate of each category consists of the list of items that were consumed by customers in that category. When a new user requires a recommendation, the recommender system computes the category to which he/she is closest, and then produces as a recommendation

on the corresponding list of items. Examples for such systems that use demographic based recommender systems include: Grundy's system [78], and Lifestyle finder [79].

Hybrid approaches are utilized by combining the mechanisms presented in the previous sections, while they are rarely used in isolation [66]. For example, Amazon adds to its collaborative filtering mechanisms additional criteria such as other books by the same author, or other books in the same category. Most recommender systems share several properties from different categories. In this way, most recommender systems use hybrid approaches to create the list of recommended items [80]. Hybrid approaches can be implemented in several ways:

- Make collaborative based, content based, knowledge based, and demographic based recommendations and then combine them.
- Add content, knowledge, and demographic based capabilities to the collaborative filtering approach (or vice versa).
- Unify all methods into one model.

2.3.3.2 According to the Network Structure

According to the network structure of the system, the recommender systems can be divided into:

- Centralized.
- Decentralized.
- Hybrid.

Centralized recommender systems store users' profiles to a central server which performs all of the required analysis on these profiles [66]. Almost all current commercial recommender systems are centralized like: Google, Amazon, Pandora, IMDB, and FilmAffinity. In this way, the service provider of the recommender system can easily

monetize the recommendations and charge for them. Centralized recommender systems have the following characteristics:

- Central database of users and items' profiles, where every piece of data in the system is stored in it. The quality of recommendations is heavily dependent on the correctness of this data and similarity metrics.
- Users obtain recommendations from a central entity and provide their feedback about recommendations through this entity.
- This model has many security issues based on the fact that users have to trust in a central entity to act reasonably as expected without any misuse of their private information that they provide to receive these recommendations.

Decentralized recommender systems contain a set of equal peers and there is no central service offered by any single node. In this way, the profiles of items and users are organized and managed in several peers of the network [66]. There is not a single point to get a recommendation. Instead, users ask their neighbours to form groups in order to get referrals. These groups are formed according only to the necessities of receiving referrals and maintain the function of the recommendation process. This type of recommender system is suitable for social networks which take the possession of two characteristics:

- Every node (peer) in these networks is equal to any other node.
- Lack of a central or traditional server.

Services and data needs to be replicated in order to cope with the structure of this recommender system, such that any peer on the network will be able to offer the recommender service [67]. Moreover, this type of recommender system is suitable for users in mobile environments due to the dynamic nature of this recommender. This type of recommender system is inherited from P2P network coverage where none has a special significance to maintain the network. An example of such systems is Napster [81].

Hybrid recommender systems appeared to handle the flaws in the decentralized recommender systems. With the current decentralized recommender systems, users create links not only to let the network work as expected, but also to improve the recommendations mechanisms [67]. These hybrid recommender systems make use of the social structure that is created within the network, such that they first organize users in neighbours or groups that share interests, where a centralized node in each group handles the recommendations in a way that is similar to the collaborative filtering model, and then each centralized node offers the recommendations based on the profiles of users in their clusters.

2.3.3.3 According to the Security

Security is an essential concern when designing any system. According to[69], an attacker might attack the recommender systems for the following objectives:

- Exposing personal private information about single or multiple users. This privacy risk is associated with any user sending his/her profile to a recommender service. This risk increases with the centralized recommender systems since there is one entity that controls users' profiles. However, this risk decreases with distributed recommender systems as users control which data to send for each recommendation process.
- Analysing the users' behaviours in order to learn patterns that exist within their personal private profiles.
- Injecting bias for certain items or inserting fake items into the system, in order to increase the chance of getting them to be recommended. This attack is called a shilling attack [82] and it can be performed from the centralized recommender or from any users of the system.
- Sabotaging the whole system so that the system will be unavailable to its intended users. As a result, no one will receive any recommendations. This attack is called a Denial-of-service attack [83].

According to the security that recommender systems can offer to their users, the recommender systems could be divided into:

- Unsecured recommender systems
- Privacy aware recommender systems

Unsecured recommender systems are those systems in which users' profiles are stored and handled in a readable form during the recommendation process. Moreover, the final recommendations are known by everybody in the system. Nearly all the previously mentioned systems like Google and Amazon are unsecured recommender systems. It should be mentioned that securing the communication with a central recommender server does not mean that the recommender system is secure. In this case, the central recommender server knows everything about its users, and the users have to trust that the central recommender is not going to misuse the information in their profiles.

Privacy aware recommender systems are those systems in which the privacy of users' profiles is protected during the recommendations process. Moreover, the users receive private recommendations and only some of the participants of the recommendation process are aware of that. There are many solutions in the literature to achieve privacy aware recommender systems. The work in [84] was the first proposal to attain this. It considers a scenario in which a centralized recommender system generates recommendations using the collaborative filtering approach. Users remove some selected parts from their profiles before sending them to the recommender. The recommender is able to attain recommendations because it was able to predict to some extent the missing parts. Attackers cannot learn the original ratings from the protected ones, but users can decide if their original ratings are included in the model using zero knowledge protocols. In this way, there is no external entity that has access to the private profile of a user. In [85], a privacy preserving approach based on peer-to-peer techniques is proposed using users' communities, where the community will have an aggregate user profile representing the

group as a whole but not individual users. Personal information will be encrypted and the communication will be between individual users but not servers. Thus, the recommendations will be generated at the client side.

In [86], a theoretical framework to preserve the privacy of customers and the commercial interests of merchants is proposed. Their system is a hybrid recommender system that uses secure two party protocols and public key infrastructure to achieve the desired goals. In [87, 88] another method is suggested for privacy preserving on the centralized recommender systems by adding uncertainty to the data using a randomized perturbation technique while attempting to make sure that the necessary statistical aggregates such as the mean don't get disturbed much. Hence, the server has no knowledge about true values of individual rating profiles for each user. They demonstrate that this method does not significantly decrease the obtained accuracy of the results. But recent research work in [30, 31] pointed out that these techniques don't provide the levels of privacy as was previously thought. In [31], it is pointed out that arbitrary randomization is not safe because it is easy to breach the privacy protection it offers. They proposed random matrix based spectral filtering techniques to recover the original data from perturbed data. Their experiments revealed that in many cases, random perturbation techniques preserve very little privacy. Similar limitations were detailed in [30]. Storing user's rating profiles on their own side and running the recommender system in a distributed manner without relying on any server is another approach proposed in [89]. The authors proposed transmitting only similarity measures over the network and keeping users rating profiles secret on their side to preserve privacy. Although this method eliminates the main source of threat against user's privacy, it requires higher cooperation among users to generate useful recommendations. Some authors explored the data-mashup of a different database to provide a joint recommendation which may be more accurate than any of the recommendations provided by the individual systems. These authors acknowledged the privacy problem that systems must solve while joining databases, and explored k-anonymity solutions [90] for that purpose.

2.4 Evaluation of Privacy Solutions

Issues concerning data privacy have emerged globally. In this section, different technological and legislative solutions for preserving data privacy will be investigated. A number of successful technological solutions which have been proposed to obtain valid results while maintaining privacy safeguards will be enumerated in the following sections. These techniques were classified into four major categories: data partitioning techniques will be outlined in Section 2.4.1, and data obfuscation techniques, group based techniques, and data restriction techniques will be described in Sections 2.4.2, 2.4.3, and 2.4.4 respectively. Finally, a brief discussion on the legislative efforts based on OECD principles will be presented in Section 2.4.5.

2.4.1 Data Partitioning Techniques

Data partitioning techniques have been applied to some scenarios in which the required datasets are distributed across a number of sites, with each site willing to share only the analysis results, not the source data. In such cases, the data can be either partitioned horizontally or vertically [91]. In a horizontal partitioning, different entities are described with the same schema in all partitions. However, in a vertical partitioning the fields of the same entities are split across the partitions. The current solutions can be further categorized into cryptography-based techniques which will be addressed in section 2.4.1.1 and model-based techniques that will be described in section 2.4.1.2.

2.4.1.1 Cryptography-Based Techniques

In the context of PETs over partitioned data, cryptographic techniques were proposed to solve problems of the following nature: two or more parties want to conduct a computation based on their secret inputs. The main requirement is how to conduct such a computation so that no party knows anything except its own input and the final results. This issue was entitled a "secure multi-party computation" problem (SMPC) [92]. Secure multi-party

computation (SMPC) is an area of cryptography that deals with the realization of distributed tasks in a secure manner. The definition of security can have various angles, such as protecting the privacy of the data or computation against malicious attacks [92]. The idea behind secure multi-party computation was introduced in [93]. The paper introduces a technique that enables the implementation of any probabilistic computation between two participants in a secure manner. Later, the concept of using secure multi-party computation was introduced in [94]. In this model, two parties owning confidential databases wish to run a predictive analysis algorithm on the union of their databases without revealing any unnecessary information. In particular, this paper focuses on the problem of decision tree learning and uses ID3, a popular and widely used algorithm for this problem. The training set is distributed between two parties. This approach treats PETs as a special case of secure multi-party computation, and not only aims at protecting individual privacy but also tries to preserve leakage of any information other than the final result. The solutions presented in [95, 96] aim at extracting globally valid results from distributed data without revealing information that compromises the privacy of the individual sources. The solution presented in [97] proposed using secure multi-party computation to conduct *k*-means clustering when different sites contain different attributes for a common set of entities. Each site has information for all of the entities for a specific subset of attributes. This work ensures reasonable privacy while limiting the communication cost. More recently, a new approach was introduced in [98] to address protecting medical data in the decision-making analysis. An algorithm is proposed to protect data before the analysis process takes place. The algorithm encrypts not only the attribute values but also the attribute labels.

Secure multi-party computation is reversible, thus allowing the results of the models to be translated back to the readable form, but only by the database owner. Secure multi-party

computation can be very inefficient and heavy in terms of communication complexity when the inputs are large and a complicated function is used.

2.4.1.2 Model-Based Techniques

Model-based techniques are designed to perform distributed analysis tasks, such that each party shares just a small portion of its local model that is used to construct the global model. The solution presented in [99] addresses privacy-preserving frequent item-sets in distributed databases. Each site sends its frequent item-sets to a combiner that finds the globally frequent item-sets based on the local models. It is shown that the global model generated is accurate and the communication cost requires only one round of message passing around the sites and one reduction operation to aggregate the final results. Another solution presented in [100] relies on Expectation Maximization (EM) based algorithms. The intuition behind this approach is that, rather than sharing parts of the original data, appropriate generative models are built at each local site, and then the parameters of such extracted models are transmitted to a central node. This central node generates artificial samples from the underlying distributions using Markov Chain Monte Carlo techniques. This approach achieves high quality distributed clustering with acceptable privacy loss and low communication cost.

2.4.2 Data Obfuscation Techniques

These techniques modify the original values of a database that needs to be shared, and in doing so, privacy is ensured. The transformed database is made available for analysis and must meet privacy requirements without losing too much from the accuracy of the results. In general, data obfuscation techniques focus on finding an appropriate balance between privacy and accuracy. Methods for data obfuscation include uncertainty addition techniques which will be described in Section 2.4.2.1, and space transformation techniques which will be explained in Section 2.4.2.2.

2.4.2.1 Techniques for Introducing Uncertainty

In statistical databases, techniques to add uncertainty to the original data can be used to protect individuals' privacy, but at the expense of disclosing some parts of the data, releasing data with less utility, and introducing biases into query responses [101]. However, for different services the major requirement of a security control mechanism (in addition to protecting the privacy) is not to ensure precise and bias-free statistics but rather to preserve the high-level descriptions of knowledge discovered from large databases [101]. Thus, the idea behind these techniques is that some noise (e.g., information not present in a particular tuple or transaction) is added to the original data to prevent the identification of confidential information relating to a specific individual. In different events, noise can be added to confidential attributes by randomly shuffling the attribute values to prevent the detection of some patterns that are not supposed to be found. Furthermore, we can classify these techniques into three categories:

- *Data swapping techniques:* These techniques replace the original database with a new one that has the same probability distribution. Such techniques are suitable for protecting privacy in knowledge discovery based services. The idea behind data swapping is that it interchanges the values in the records of the database in such a way that statistics about groups (e.g., frequencies, averages, etc.) are preserved. The method proposed in [102] was designed for protecting the statistics of the released data for analysis. In this approach, a new dataset which is released to the services is a perturbed version of the original data set. It has also been shown that it is possible to balance statistical precision against the security level by choosing to perform the swapping in groups of records rather than in a small number of records.
- *Data perturbation techniques:* These techniques distort the data to protect individuals' privacy by introducing an error (noise) to the original data. The noise is used to generate the new (distorted) database which is subjected to analysis. Service providers

should be able to obtain valid results (e.g., patterns and trends) from the distorted data. As opposed to statistical data analysis, services do not aim at obtaining a definite, unbiased statistical test that answers with a probabilistic degree of confidence whether the data fit a preconceived statistical model. The work presented in [103] aims at building a decision-tree classifier from training data in which the values of individual records have been perturbed by adding random values from a known distribution. The distribution of the resulting data appears different from the distribution of the original data. The authors proposed a novel reconstruction procedure to accurately estimate the distribution of the original data values in order to build classifiers whose accuracy is comparable to the accuracy of the classifiers built with the original data.

- *Data randomization techniques:* These techniques allow one to discover the general patterns in data-sets within a specific boundary error, while conserving the original data values. Like data swapping and data perturbation techniques, randomization techniques are designed to find a good compromise between privacy protection and knowledge discovery. A framework for mining association rules from transactions consisting of categorical items was proposed in [104]. In this approach, the data are randomized to preserve the privacy of individual transactions. The basic concept in this work is to replace some items in each transaction by new items not originally present in this transaction. As a result, some true information is elicited and some false information is introduced to obtain reasonable privacy protection. A new data randomization technique has been applied to Boolean association rules [105]. The idea is to modify data values such that reconstruction of the values for any individual transaction is difficult, but the rules learned from the distorted data are still valid. Although this framework provides a high degree of privacy, mining the distorted database can be more expensive in terms of both time and space as compared to mining the original database. Considering that randomization is an efficient approach for PETs, some effort

has been made to optimize the trade-off between knowledge discovery and the protection of individuals' privacy.

2.4.2.2 Data Transformation Techniques

Data transformation techniques aim at protecting the underlying data values subjected to analysis without jeopardizing the similarity between objects under analysis. Thus, a data transformation technique must not only meet privacy requirements but also guarantee valid analysis results. A geometric data transformation method was proposed in [106] to distort numerical attributes by a combination of translations and rotations, the viability of using either a specific or the combination of these transformations for protecting privacy was extensively studied. The key finding was that by transforming a data matrix by rotations and translations, one would attain both accuracy and a reasonable level of privacy. A more accurate investigation on using geometric transformation is presented in [107]. In particular, it is shown that distorting attribute pairs in a database by using only rotations is a promising approach to protect data privacy without jeopardizing the similarity between data objects under analysis. This technique is similar to obfuscation since the transformation process makes the original data difficult to perceive or understand, and preserves all the information for analysis.

2.4.3 Grouping Based Techniques

There are several methods thought to have been developed by researchers in the database community that handle tuples in a “group-based” mode, using information about specific tuples globally to transform them in a way which preserves specific privacy metrics. These modified records can then be published without fear of reconstruction by attacks such as those described above, even in the presence of certain kinds of domain knowledge linked against the released dataset. However, a key problem with these group-based methods is the suitability of the transformed records for different analysis tasks, and the general utility of

the published data. In many of the subsequent methods, a general assumption is assumed that specific fields of a record contain quasi-identifier attributes that uniquely identify an individual associated with the record, as well as sensitive attributes that must not be linked to the individual by an untrusted third party. Grouping-based techniques can be classified into three variants as listed below. These techniques differ in their own objective metrics but rely on achieving the final state where *k* records look exactly the same.

- *k-Anonymity:* It is considered the first and most famous heuristic-based approach developed for anonymisation which was introduced by [108, 109]. This framework is predicated on the fact that when publishing anonymised database records to external parties, eliminating identifier fields such as a name or social security number may be insufficient to stop attackers from linking the records against public data sources. Each released record in the dataset is associated with an individual, and contains values that are *personal*, which should not be directly linked to that individual after anonymisation. The record also contains some *quasi-identifier* fields, which when taken in combination and linked to some external source yield the connection between the record and the individual. The concept of *k*-anonymity conceptualizes a user specified privacy level that must be obtained by transforming the records before the data can be published. To be *k*-anonymous, a data release must be modified such that every combination of quasi-identifiers can be indistinguishably matched to at least *k* individuals. The most common operators are suppression (removing all or part of the values of a field) and generalization (using some predetermined generalization tree). In general, given some cost function for suppressing or generalizing values, it is *NP*-hard to compute the minimal cost *k*-anonymisation of a given data set [110, 111].
- *l-Diversity:* A major drawback of the *k*-anonymity metric introduced in [109] is that it can leave an anonymised data set open to so-called "homogeneity attacks". The basis of the attack is that *k*-anonymisation algorithms can potentially create equivalence classes

of *k* anonymised records that are each associated with the identical *sensitive* attribute value. To overcome this deficiency, the authors in [112] introduced the concept of *l*-diversity, where they presented an algorithm that prejudices the search for a *k*-anonymous transformation to those in which at least *l-sensitive* attribute values appear in each equivalence class.

- *t-Closeness:* Authors in [113] demonstrated that even after anonymising to produce an *l*-diverse transformed data set, the distribution of sensitive attribute values might be changed in such a way that it reveals some information about them. This is known as the *similarity attack;* since it exploits the similarity of sensitive attribute values. To defeat attacks such as these, authors proposed the *t*-closeness criteria, which stipulates that each *k*-anonymous equivalence class should contain sensitive attribute values that are distributed as closely as possible to the original overall data set.

2.4.4 Data Restriction Techniques

Data restriction techniques focus on limiting the access to certain results through either the generalization or suppression of information or by preventing the extraction of specific patterns that are not supposed to be found. Such techniques can be classified into two categories, blocking-based techniques that will be described in Section 2.4.4.1 and hiding-based techniques which will be presented in Section 2.4.4.2.

2.4.4.1 Blocking-Based Techniques

Blocking-based techniques aim at hiding some sensitive information when data are shared with external services. The private information includes sensitive results that must remain private. Before releasing data for analysis, data owners must consider how much information can be inferred or calculated from large databases and must look for ways to minimize the leakage of such information. The work presented in [114, 115] investigates the need for developing secure policies to minimize or eliminate the threat introduced by

knowledge extraction algorithms in the context of analytical services. It is shown that from unclassified data, one is able to infer confidential information that is not supposed to be disclosed. This work was later extended in [116] for privacy-preserving association rule mining which is a new methodology that was proposed to hide knowledge in relational databases and to control the unauthorized disclosure of directed and undirected inferences.

2.4.4.2 Hiding-Based Techniques

Unlike blocking-based techniques that hide sensitive information by limiting or replacing some items or attribute values with unknowns, hiding-based techniques hide sensitive information by strategically suppressing some values in the databases or generalizing them to protect privacy in results. These techniques can be categorized into two major groups: data-sharing techniques and pattern-sharing techniques. In the former, the hiding process acts on the data to remove or hide the group of sensitive association rules that contain sensitive knowledge. In order to do so, a fraction of the transactions which contain the sensitive rules have to be modified by deleting or adding one or more items. In the latter case, the concealing algorithm acts on the rules extracted from a database, instead of on the data itself. The algorithm removes all sensitive rules before the sharing process.

The idea behind data-sharing techniques was first introduced in [117]. The authors considered the problem of limiting disclosure of sensitive rules by hiding some frequent item-sets from large databases with as little impact on other non-sensitive frequent item-sets as possible. In [116], the authors investigated confidentiality issues related to a broad category of association rules and proposed some algorithms to preserve the privacy of such rules above a given privacy threshold. Although these algorithms enhance privacy, they are expensive in terms of execution time since they require numerous scans over the database.

Regarding pattern-sharing techniques, one approach that falls into this category was introduced in [118]. This method addresses the sharing of association rules between two or

more parties. This method is composed of a concealing algorithm for protecting sensitive knowledge before sharing association rules. A framework was proposed in [119] to transform an original database into a new one by using generalization and suppression to satisfy some privacy constraints. This work also introduces some metrics to quantify information loss in the transformed database. The data transformation problem is solved by using an algorithm to optimize the appropriate metric. This work considers the trade-off between privacy and information loss.

2.4.5 Fair Information Practice Principles for Privacy

In this section, the privacy principles established by the Organization for Economic Cooperation and Development (OECD) are highlighted in Section 2.4.5.1, and the implications of utilizing those fair information practice principles in the context of our proposed privacy enhancing framework will be analysed in Section 2.4.5.2. Section 2.4.5.3 will highlight the boundaries that are utilized in designing privacy enhancing frameworks.

2.4.5.1 The OECD Privacy Principles

The Organization for Economic Co-operation and Development (OECD) [34] formulated sets of principles for fair information practice that can be considered as the primary components for the protection of privacy and personal data. A number of countries have adopted these principles as statutory law, in whole or in part in order to govern the data that customers provide for third party services operating at remote sites. These principles can be described as follows:

- Collection limitation: Data collection and usage for a remote service should be limited only to the data that is required to offer an appropriate service.
- Data quality: Data should be used only for the relevant purposes for which it is collected.

- Purpose specification: Remote services should specify up front how they are going to use data and users should be notified up front when a system will use it for any other purpose.
- Use limitation: Data should not be used for purposes other than those disclosed under the purpose specification principle without user consent.
- Security safeguards: Data should be protected with reasonable security safeguards (encryption, secure transmission channels, etc.).
- Openness: The user should be notified upfront when the data collection and usage practices started.
- Individual participation: Users should have the right to insert, update, and erase data in their profiles stored at remote services.
- Accountability: Remote services are responsible for complying with the principles mentioned above.

2.4.5.2 The Implications of OECD Principles in Designing Privacy Enhancing Frameworks

In the next section, the research work in [120] is presented that classifies the implications of OECD principles with respect to privacy frameworks. Their suggestions will be used in order to state which principles should be considered as a norm in designing our privacy enhancing framework:

- Collection Limitation: This principle is too general to be applied in our privacy enhancing framework. As a result, we decided to leave this task for the user to determine a sensible realization for the notion of very sensitive data. Moreover, the users are responsible for making their data public or private by employing privacy preferences languages to specify rules or levels for releasing their data such that a conscious automatic choice can be made about which group gets to see what. By catering to the second boundary, it gives the end-users the choice to join a peer-group,

using an anonymous network or leaving the recommendation process, where the users can join a peer-group only with trusted end-users or their friends.

- Data Quality Principle: Most of the privacy enhancing frameworks assume that the data is in an appropriate form to be processed by obfuscation or anonymization techniques. However, data cleaning methods can be utilized locally to handle imprecision and errors in data before any concealment process. We mitigated this principle by selecting two common types of erroneousness in the users' data, which are incomplete users' profiles and outliers. We then proposed a set of concealment algorithms which take into consideration pre-processing the incomplete user profiles and handling outliers on these profiles. Other types of deviations should be investigated in future research. Meanwhile, we left the task of handling other erroneousness to the user, in order to maintain an accurate profile for the recommendation process and to facilitate a straightforward concealment process.
- Purpose Specification Principle: This principle is relevant for our privacy enhancing framework. Users should be well informed at the outset prior to the collection and processing of their information.
- Use Limitation Principle: This principle is relevant for our privacy enhancing framework and related to the previous principle. The collected information from users must be used only for the purpose that was disclosed at the time of collecting this information.
- Security Safeguards Principle: This principle is relevant for our privacy enhancing framework but related in general to data security. We have mitigated this principle by associating two profiles for each user; one is a local profile in a plain form that is stored locally in his/her machine and the other is a public profile in a concealed form that is stored remotely at the remote service. This ensures that personal users' data is protected from unauthorized users.

- Openness Principle: This principle is relevant for our privacy enhancing framework. Users should know what data about them have been gathered and processed. However, most of the social recommender services do not disclose the logic behind the scene due to intellectual property issues. Our framework enables the user to decide either to join or not to join a certain recommendation process and also to control what data to release within a certain recommendation process.
- Individual Participation Principle: This principle is relevant for our privacy enhancing framework. Users are aware that the generated referrals are related to their released data. Users can challenge the value of the generated referrals and decide either to participate or not. Therefore, there should be a certain mechanism to carefully outline the weight of this principle to the users.
- Accountability Principle: This principle is irrelevant for our privacy enhancing framework. Remote services should inform users about the policies related to the usage of the generated recommendation model including the consequences of abusing the collected data. This principle is too general in scope or area to be utilized for PETs.

Based on the outline we declared above, we categorized the OECD principles into two groups according to their influence on the context of PETs:

Group 1 consists of those principles that should be considered as design principles in our privacy enhancing framework, such as data quality, purpose specification, use limitation, security safeguard, openness and individual participation.

- Group 2 involves some principles that are too general or irrelevant in PETs. Some of those principles depend on the applications where PETs are needed, and their effects should be understood and carefully evaluated depending on these applications.

The principles categorized in group 1 are relevant in the context of our collaborative privacy framework and are fundamental for further research, development, and deployment of privacy enhancing frameworks.

2.4.5.3 The Boundaries of Designing Privacy Enhancing Technologies

The boundaries and content of what is considered private differ among cultures and individuals, but share basic common themes. Inspired from the work in [121], we summarized the challenges for developing a privacy enhancing technique as boundaries and for each boundary, we describe a tension which the boundary has to face. These boundaries are as following:

- The disclosure boundary (privacy and publicity) - this can be defined as a tension between the privacy and publicity. The user has to decide what to keep private and what to make public.
- The identity boundary (self and other) - the users need to decide which identity to disclose to whom. This is a tension between different identities that a user might have.
- Temporal boundaries (past, present and future) - this is a tension on the time aspect. What is not private in the past might become so in the future and vice versa, and when the information is being persistent much of the actions done in the past cannot be undone.

As we presented in the appendices, our contributions address the first two boundaries. The end-users have the choice to make their data public or private by employing privacy preferences languages to specify rules or a level for releasing their data such that a conscious automatic choice can be made about which group gets to see what. Other options catering to the second boundary include: giving the end-users the choice to join a peer-group, use an anonymous network or leave the recommendation process, where the users can join a peer-group only with trusted end-users or their friends. However, the temporal boundary is not really addressed in our current work, but we plan to address it in future work.

Chapter 3

Research Summary

The research work described in this thesis is believed to make a significant contribution towards distributed clustering techniques and introduces a novel collaborative privacy framework with stochastic techniques for social recommender services. The different parts of the contributions will be described in this chapter. Section 3.1 will present a novel distributed clustering algorithm, while Section 3.2 will depict the attack model for social recommender services. Section 3.3 will present our novel collaborative privacy framework for social recommender services. Sections 3.4, 3.5, 3.6, 3.7, and 3.8 will present five applications of our collaborative privacy framework, which are privacy aware data mash-ups for IPTV recommender service, privacy aware recommender service for IPTV content providers, private community discovery & recommendation service, privacy aware mobile jukebox recommender service, and privacy aware location based recommender service respectively. Finally, Section 3.9 will present the answers to the research questions that were introduced in Section 1.5.

3.1 A Distributed Clustering Algorithm

In this thesis, Article II will present an efficient distributed clustering algorithm (DLC) that bears in mind the privacy issue. The model building mechanism in DLC is based on two mechanisms that use only the statistics (field function) about the data without the need to gather all of the original data from each site. DLC requires only three parameters by following two steps which are presented below: (1) local learning and analysis step *(LLA)* and (2) distributed clustering step *(DC)*.

Local Learning and analysis Step (LLA)

LLA is an elementary pre-processing and essential step in all of our proposed PETs. It was developed based on research work in Article I, which investigates a method for selecting a subset of relevant features from which a clustering model is built. The feature weight in the dataset is estimated using a weight factor $FWA(a_i)$

$$FWA(a_i) = \frac{\left(\sum_{j=1}^{n} E(a_j)\right) - E(a_i)}{\sum_{j=1}^{n} E(a_j)}, \forall\ 1 \le i \le n\ \&\&\ E(a_j)\ \textit{is the estimation index} \quad (3.1)$$

LLA can be utilized for detecting dense regions and outliers from the dataset using an influence function that is proposed in [122]. We used a Gaussian influence function as an indicator that is calculated for nearest neighbours. All of the other points can be neglected without causing considerable error. We calculated the field function for a point as a summation of the influence in its nearest neighbours. A detailed description of *LLA* is presented below.

Distributed Clustering Step (DC)

The DC algorithm uses the single link (slink) algorithm in [123], but with some modification to estimate the merge error based on [2]. It uses the membership function in [124] with modification based on the field function notation. The algorithm takes one input which is the least error threshold (LET). DLC steps 1 and 2 are illustrated below.

A privacy preserving version of the DLC algorithm using cryptographic protocols was presented in Article II with an application of the hospitals' collaborations.

DLC Step 1 *Local Learning and Analysis (LLA)*

Require: local parameter α, threshold β

1. *Influence of x on y ($x, y \in F^d$):* $f^x_{Gauss}(y) = e^{-\frac{d(x,y)^2}{2\sigma^2}}$.
2. *The field function for a point:* $f^D_{Gauss}(y) = \sum_{i=1}^{k} e^{-\frac{d(x,y)^2}{2\sigma^2}}$ *where k are the nearest neighbors for y.*
3. *Calculate outlierness degree factor (ODF) [1]:* $ODF(s_i) = \frac{f^D(y)/k}{f^D(s)}$, **if** $ODF(s_i) \gg 1$, s_i *is local outlier.*
4. *Based on local parameter α, $\forall\ f^D(y) \geq \alpha$, y is candidate local core point. Note that α value should be a point that shows variation in densities.*
5. *Based on influence and field functions, it calculates initial cluster density [3] as follows:*
   $D_{cluster_i} = size_{cluster_i}/D_{Av}$, $D_{Av} = f^D(y)/k$
6. *Check each data point* **if** *it will increase or decrease the density of the clusters when it joined or left clusters respectively within fixed threshold β, and then calculated:*
   $D_{cluster_e} = size_{cluster_e}/D_{Av}$, $D_{Av} = \sum_{i=1}^{n} f^x(y)/n$
7. **If** $D_{cluster_e} > D_{cluster_i}$
   The clusters that amplify density gain when data points joined it; are chosen as candidate clusters from the density perspective, **else** *keep the current.*
8. *Recalculate the influence and field functions to each $cluster_i$'s member; and the highest density point will be local core point.*

DLC Step 2 *Distributed Clustering (DC)*

Require: Error threshold LET

1. *Calculate the summation of all field functions for each point in all sites and store it in a sorted table.*
2. *All sites agree on common α range to classify high density points.*
3. *Select the highest density point to be the starting point for the current cluster.*
4. **If** *the same point in all sites ⊂ dense point's cluster neighbors*
   Then *Expand the current cluster by adding this point*
   Else *Assign each point to its nearest dense point according to:*
   $$m(y_j|x_i) = \frac{\|x_i - y_j\|^2}{\sum_{j=1}^{k}\|x_i - y_j\|^2}$$
5. *Start a new cluster and* **repeat** *steps 2 to step 4* **until** *all points are clustered.*
6. *Assign outliers based on step 4. To the nearest cluster.*
7. *According to [2] merging two clusters based on least error:*
   $error\ (y_i \cup y_{i+1}) = \frac{y_i.w_i \times y_{i+1}.w_{i+1} \times d(y_i, y_{i+1})^2}{y_i.w_i + y_{i+1}.w_{i+1}}$
   the membership function is calculated as follows:
   $$w_i(y_i|x_1, x_2, .. x_o) = \sum_{n=1}^{o} \frac{\| f^{x_n}(y_i)\|^2}{\sum_{j=1}^{k}\|x_i - y_j\|^2}$$
8. **Repeat** *step 7, until the error* $>$ *LET*.

Privacy in Pervasive Healthcare Service

Article XIV presented the guidance for developing personalized health systems (pHealth) which comply with the requirements of immerse medical devices in daily patients' activities without changing their life activity. pHealth scheme was presented (as shown in Figure 3.1) which address technological limitations in order to extend dynamic user participation.

Utilizing patients' health profiles, a private monitoring application for the analysis of ECG data within a personalized healthcare system was given in Article XV. This scenario requires capabilities of analysis of distributed profiles. The proposed DLC algorithm was utilized on a real dataset to diagnose the cardiopathy condition of ECG patients' signals. The *LLA* step was used to gather statistics about ECG data then a clustering model was

built based on these statistics to enable a private analysis of ECG data in order to discover specific cardiovascular disease patterns. An overview of this monitoring application is presented in Figure 3.2.

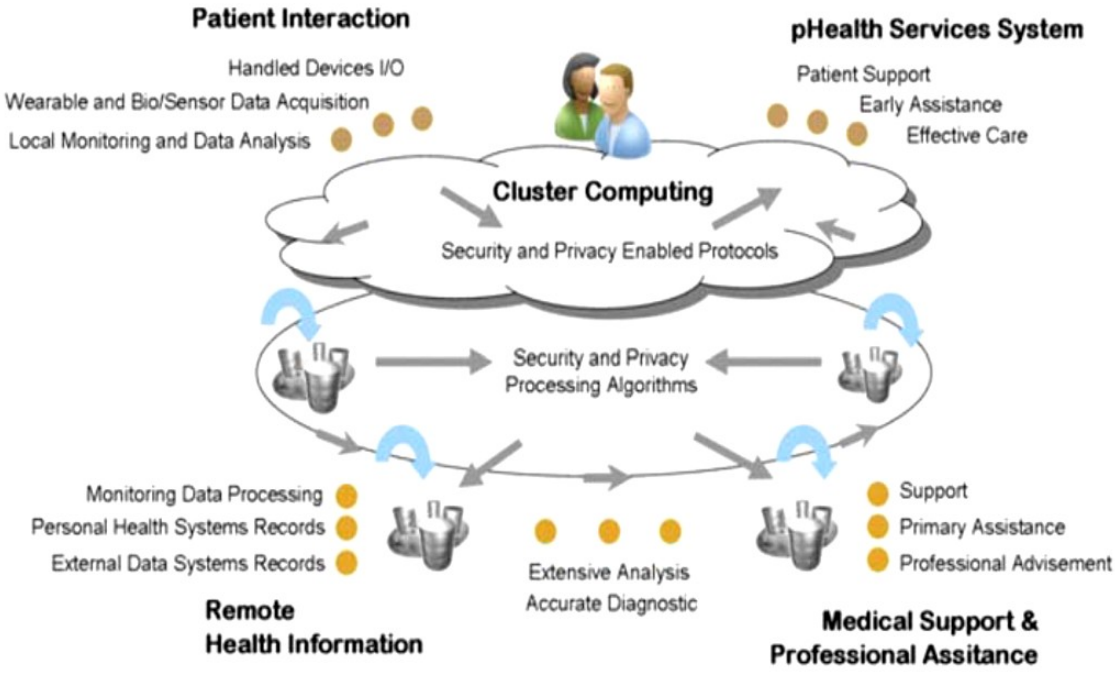


Figure 3.1: Personalised Healthcare Services Schema

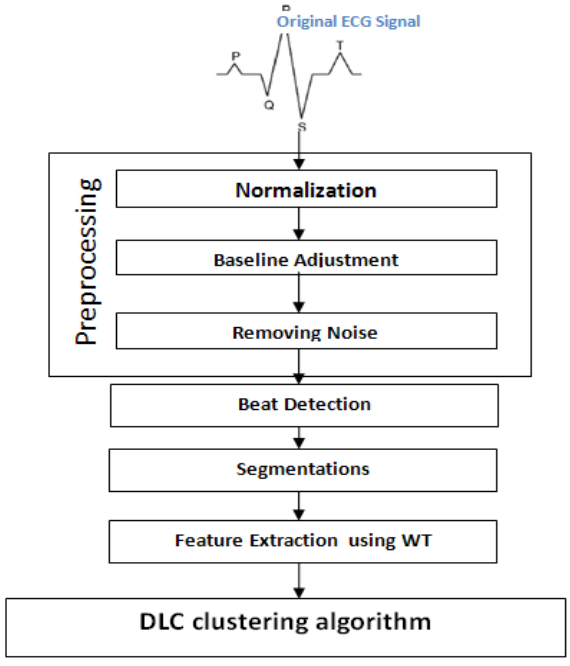


Figure 3.2 Personalized Medical Application for Early Cardiovascular Diagnostics

Finally, Article XVI presented a collaborative platform called a distributed platform of health profiles (DPHP) for personal health profiles that enable individuals or groups to benefit from their personal health profiles. It stores user's personal health profiles in a non-proprietary manner which will enable healthcare providers and pharmaceutical companies to reuse these personal health profiles in parallel in order to maximize effort where users benefit from each usage for their personal health profiles. DPHP utilizes a modified version of the *LLA* step to facilitate the selection of appropriate data aggregators and evaluating their offered data in an autonomous way. We utilized PETs proposed in Article VIII to preserve the privacy of the merged profiles from multiple sources involved in the data aggregation. The reason for choosing these PETs is their ability to preserve the aggregates in the dataset to maximize the usability of information in order to attain accurate machine learning clustering.

Algorthm 1: *Modified LLA Clustering Algorithm*

Require: Initial values: $X_i(i = 1 \dots n)$, Number of categories: k
1. **Select** *any values* $X_{i1}, X_{i2}, X_{i3}, \dots X_{ik}$ **from** X_i *randomly*
2. **Set** *an initial starting category* $Y_j = X_{ij}(j = 1 \dots k)$
3. **Do until** *the group member is stable*
For each $X_i(i = 1 \dots n)$
If $X_i \in [Y_j, Y_{j+1}]$
$D_1 = e^{-\frac{d(X_i, Y_j)^2}{2\sigma^2}}$
$D_2 = e^{-\frac{d(X_i, Y_{j+1})^2}{2\sigma^2}}$
If $D_1 < D_2$ *then* X_i *is in the cluster (category) of* Y_j
Else X_i *is in the cluster (category) of* Y_{j+1}
End if
End If
End for
$Y_j =$ *the average of cluster* $Y_j(j = 1 \dots k)$
End Do

Evaluation Results

In order to measure the accuracy of the *DLC* algorithm in determining different heartbeat clusters, Figure 3.3 shows the relation between merge error in the DC stage and the number of clusters. As shown in Figure 3.3, the merge error *(LET*) decreases, which indicates that only equivalent heartbeat clusters are being merged. Moreover, in order to evaluate the performance of the proposed algorithm, two error metrics defined in [125] were used. The first metric is clustering error (CR) which is the percentage of heartbeats in a cluster that do not correspond to the class of such a cluster and the second metric is the critical error (CIE) which is the number of heartbeats in a class that do not have a cluster and are therefore included in other's classes' clusters. The relation between the different number of clusters and the values of clustering error (CR) and critical error (CIE) was depicted on Figures 3.4 and 3.5.

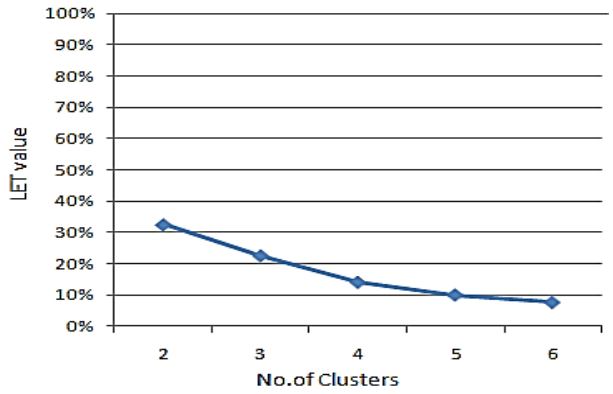


Figure 3.3: The Relation between Different Clusters and Merge Error

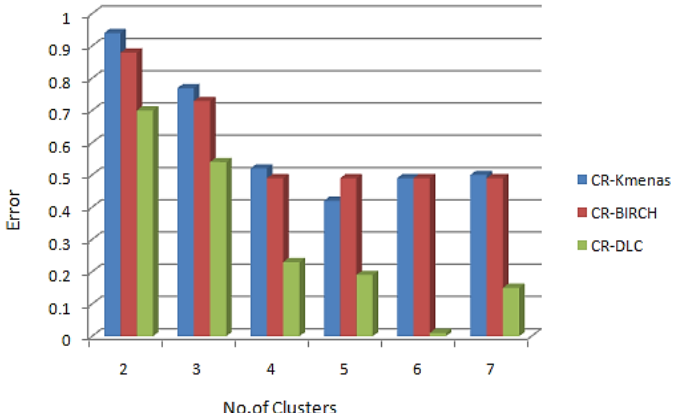


Figure 3.4: The Values of CR for Different Number of Clusters

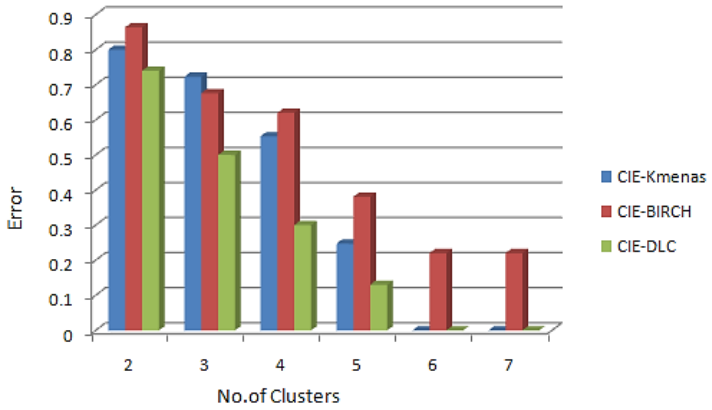


Figure 3.5: The Values of CIE for Different Number of Clusters

As seen above, both CR and CIE for the *DLC* algorithm decrease with the increase in the number of clusters until reaching correct number of clusters. Finally, comparing the obtained results from *DLC* algorithm with other clustering algorithms like BIRCH and k-means; the results show that the obtained accuracy of the results achieved using *DLC* algorithm is close to the ones obtained with these algorithms.

3.2 Attack Models for Social Recommender Services

In this research, collaborative privacy assumes that the user's profile is stored at the user side in his/her device. Usually, the user's profile suffers from the sparsity problem since it only contains preference data for a small fraction of the items that have been consumed by its user. This sparsity in preference data significantly increases the difficulty of any concealment techniques since it has an adverse effect on the attained privacy level. Unfortunately, existing algorithms are not effective when applied to sparse preference datasets. Moreover, most of the existing anonymization algorithms are built around the assumption that there are two non-overlapping value sets: sensitive values, which need to be kept private, and quasi-identifier values, which can be used by the attacker to identify individuals. While in preference data, these two sets are not disjoint; all preference information could be sensitive, and can also potentially be used as quasi-identifiers. This additional challenge requires new privacy models that need to be applicable for preference

data. The proposed two stage concealment process which is hosted within our collaborative privacy framework fall into the category of data obfuscation techniques. It is known that the attack models for data obfuscation techniques are different from the attack models for encryption-based techniques. However, no common standard has been implemented for data obfuscation so far. Existing techniques have primarily considered a model where the attacker correlates obfuscated data with data from other publicly-accessible databases or leaked datasets about the victim in order to be able to uniquely identify a user. This is especially true if the user has "not-so-popular preferences" that reveal sensitive items in his/her profile.

In this research, we assume that an adversary aims to collect preferences in user's profiles in order to identify and track users. Thus, we consider our main adversary to be an untrusted recommender service to which users send their preferences. We do not assume the social recommender service to be completely malicious. This is a realistic assumption because the social recommender service needs to accomplish some business goals and increase its revenues. The social recommender service can construct the profiles of the users based on the requests sent. Hence, the problem we are tackling has two sides. We want to detain the ability of the adversary to identify users based on a set of identifying interests and thus track them by correlating these data with data from other publicly accessible databases or leaked datasets about the victim. At the same time, we want to prevent the adversary from profiling the users through their network identity and therefore invade their privacy. Intuitively, the system privacy is high if the social recommender service is not able to reconstruct the real users' preferences based on the information available to it.

Two main threat models were employed to mimic attacks on the proposed two stage concealment process, mainly, the Graph Matching [126] and Re-identification [127] Attacks. Within the Graph Matching Attack, an untrusted social recommender service

trying to link the group profiles' data to certain users was modelled. The user profiles contain a set of item ratings γ. Upon receiving a request for a recommendation process, each user within the peer-group performs local concealment on his/her items' ratings then forwards them to the super-peer, who will aggregate all these ratings profiles of the member within the peer-group in one group profile to form γ_{ag}.The super-peer will execute a global concealment on this group profile, then will forward it to the social recommender service. Each user has a hidden set of item ratings γ_{hid} and a released set of item ratings γ_{rel} . The γ_{rel} is already in the group profile at the super-peer side. The total item ratings by all users can be represented as a bipartite graph, with each user P within a peer-group n represented as set of nodes P_n and the complete item ratings set γ. The set of edges connecting a user in $P_n$to a subset of γ defines the user's profile. The hidden graph, G_{hid}, contains the hidden items' ratings of the user while the release graph G_{rel}, is the graph built by an untrusted social recommender service provider colluding with one or more of the super-peers/members. The privacy metric proposed in [126] was employed to evaluate the attained privacy by the proposed concealment process. The metric measures the achieved privacy in the two stage concealment process using concepts of graph matching, where the untrusted social recommender service tries to match G_{hid} to G_{rel}. The value S_i represents the frequency of releasing this items' rating for different requests regarding this item's genre. The higher value for this metric implies a higher privacy level attained.

$$\text{privacy} = \frac{1}{N} \times \sum_{P_n} \frac{\sum_{i \in (\gamma_{rel}/\gamma)} \frac{1}{S_i}}{\sum_{i \in (\gamma_{rel})} \frac{1}{S_i}}$$

The Re-identification Attack measures the disclosure risk of the proposed global concealment technique as the probability of re-identifying the globally concealed group profile based upon a portion of the originally released profiles from a malicious member within a peer-group. Inside this attack, the untrusted social recommender service in collaboration with a malicious peer wants to filter out the existing collected items from a

group profile based upon a portion of the originally released items which have been disclosed by a malicious member within a peer-group in order to discover if certain preferences were released by the victim's profile. As a result, the record linkage [127] metric was employed to measure the difficulty of finding correct matches between the original preferences and its concealed version within the group profile. Given a group profile as the set $\gamma^{O} = \{\gamma_1^o, \gamma_2^o, \gamma_3^o \ldots \gamma_n^o\}$, the record linkage can be expressed as:

$$R_L = \frac{\sum_{i=1}^{n} Pr(\gamma_i^o)}{n} * 100$$

Where $Pr(\gamma_i^o)$ is the probability of a concealed rating for specific item, and it is computed as following:

$$Pr(\gamma_i^o) = \begin{cases} 0 \text{ if } \gamma_i^r \notin L \\ \frac{1}{|L|} \text{ if } \gamma_i^r \in L \end{cases}$$

Where γ_i^r is the original rating, γ_i^o , is its concealed version and L is the set of original ratings about a specific item, that has matched with the rating γ_i^o. Thus, we searched the original profile γ^r for matches with a concealed rating γ_i^o, the set of matched ratings L is searched for matches with γ_i^r based on an item. If $\gamma_i^r \in L$, then $Pr(\gamma_i^o)$ is calculated as the probability of finding γ_i^r in L. If no matches are found, $Pr(\gamma_i^o) = 0$. Articles III, IV, VII, VIII, XI and XIII present a realization of these attacks and how our proposed PETs within the two stage concealment process protect the profile privacy against these threats.

The Role of OECD Principles in Collaborative Privacy Framework

In this section, we will discuss the role of OECD principles in designing the proposed collaborative privacy approach which reduces privacy risks and facilitates privacy commitment. The proposed solution realizes privacy aware recommendations while complying with the current business model of a third-party social recommender service. The privacy obtained through the proposed collaborative privacy approach is as follows:

- Collection Method: The proposed solution attains an explicit data collection mode. Users are aware that a data collection within a recommendation process is happening and they can make a wise decision about whether or not to provide their data in this recommendation process. Privacy policies such as P3P are utilized to explain to the users how their data is going to be used. Users utilize privacy preferences in order to control what data from their profiles gets collected at each concealment level. However, formalizing such privacy preferences is not an easy task. Users need to realize various privacy issues. Additionally, users need to deduce future recommendation requests that might raise privacy concerns for his/her collected data. The user can employ an anonymous network while sending this locally concealed data to either the super-peer or the social recommender service.
- Duration: The proposed solution attains a session based collection that allows for a simpler service that does not need the storage and retrieval of users' profiles. The data related to the recommendation process is collected from the users' profiles in a concealed form. This concealed data is only feasible for the recommendation purposes. This reduces privacy concerns since minimal data will be collected and also ensures compliance with privacy laws. The concealed data is stored at the third party service in order to enhance the recommendation model and future requests. Moreover, this data by default is protected by the retention policies of data protection laws.
- Initiation: The proposed solution attains a user based recommendation. Users are the entities that initiate the recommendation process. Each user in the network is aware that a recommendation process is happening and he/she can decide whether or not to join it. The incentive for participants when joining a recommendation request includes receiving referrals regarding a certain topic in a private manner.
- Anonymity: The proposed solution attains anonymity which aids in preventing frauds and Sybil attacks. The anonymity is realized within the collaborative privacy framework using the following procedures:

a. Dividing system users into a coalition of peer-groups: each peer-group will be treated as one entity by aggregating its members' concealed data in one aggregated profile at the super-peer. This super-peer will then handle the interaction with the social recommender service. Participants within the coalition interact with each other in a P2P fashion and form a virtual topology to aggregate their data.
b. Using anonymous channels like Tor: individual participants might benefit from these anonymous channels while contacting the recommender service or other members in their coalition.
c. Utilizing pseudonyms for users: each user within the system is identified by a pseudonym in order to reduce the probability of linking his/her collected profiles' data with a real identity.

- Local Profiles: Our solution attains local profile storage. Users' profiles are stored locally on their devices (setup box, smart phone, laptop...) in encrypted form. This can guarantee that these profiles are attainable only to their owners. Furthermore, in doing so these profiles will be inaccessible to viruses or malware that may affect the user's machine to gather his/her personal data. As a result, each user will possess two profiles; one is a local profile in a plain form that is stored locally in his/her machine and is updated frequently. The other is a public profile in a concealed form that is stored remotely at the service provider and is updated periodically within each recommendation process where this user participated.
- Privacy enhancing technologies: Our solution relies on a set of machine learning cluster analysis based stochastic PETs. These PETs are to be carried out in two consecutive steps within a two stage concealment process. The proposed PETs destroy the structure in data but, at the same time, maintain some properties in it which is required in the planned recommendation. The implementation of such applications also confirmed that it is feasible to make use of and, at the same time, to protect the personal sensitive data of individuals, and to do so in an accurate way.

3.3 Collaborative Privacy Framework for Social Recommender Services

In this thesis, Articles VI, VII, VIII, X, and XIII present a collaborative privacy framework that implement a two stage concealment process. This framework utilized a set of machine learning based stochastic techniques that introduce carefully chosen artificial noise in the data so as to retain its statistical content while concealing all private information. In that way, privacy is achieved for both individual participants and groups of participants. The proposed framework was applied as a middleware which combines all of these techniques to make it possible to efficiently take advantage of this work. This middleware enables participants to be organized in a distributed topology to achieve privacy for participants with relatively low accuracy loss. However, this topological formation prevents the service provider from creating a centralized database with raw personal data from each user. It also permits a decentralized execution of a two-stage concealment process on users' personal data. This topological execution in the proposed middleware satisfies the requirements of high scalability and reduces the risk of privacy breaches. The validity of the framework is demonstrated by the implementation and evaluation of the proposed solution within a set of important innovative applications. A general overview of the proposed framework is shown in Figure 3.6

The framework demands that users be organized into peer-groups with a specific virtual topology in order to create an aggregated profile (group profile).This topology might be simple like ring topology or complex like hierarchical topology (see Figure 3.7). This ordering enables users to attain privacy by collaboration between them. Data is shared between various users within the same peer-group after it is locally concealed. The super-peer will be responsible for executing a global concealment process on the aggregated profile (group profile) before delivering it to the recommender service.

In our collaborative privacy framework, the notion of privacy surrounding the disclosure of users' preferences and the protection of trust computation between different users are together the backbone of this framework. A trust based obfuscation mechanism was applied at the participant side such that trust computation is done locally over the obfuscated participant's preferences. Utilizing a trust heuristic as input for both group formation and the obfuscation process has been of great importance in mitigating some of the malicious insider attacks described in Article XIII.

Figure 3.8 illustrates the enhanced middleware for collaborative privacy (EMCP) which in the early version was called (AMPR) components running inside the user's local device. EMCP consists of different cooperative agents. A learning agent captures the user interests about miscellaneous items explicitly or implicitly to build a rating database and meta-data database. The manager agent is responsible for coordinating between requests and different agents in EMCP, such that, the manager agent receives the request from the target user along with the P3P policy from the elected super-peer. It then forwards them to the involved agents. It also ensures that the collected preferences are the required preferences for the particular request that the user is engaged in. The local obfuscation agent implements the local concealment process to achieve user privacy while sharing his/her preferences with super-peers or a private recommender service (PRS). The trust agent calculates the approximated interpersonal trust between its host and the target user based on their shared preference. It is done in a decentralized fashion using the entropy definition proposed in [128]. The encryption agent is only invoked if the user is acting as a super-peer in the recommendation process. It executes global concealment on the aggregated profile (collected profiles from the peer-group). The two stage concealment process acts as wrappers that conceal preferences before they are shared with any external social recommender service. Within the context of the work proposed in Article VI and Article

VII, the encryption agent utilized an additively homomorphic cryptosystem as a global concealment process on the aggregated profile.

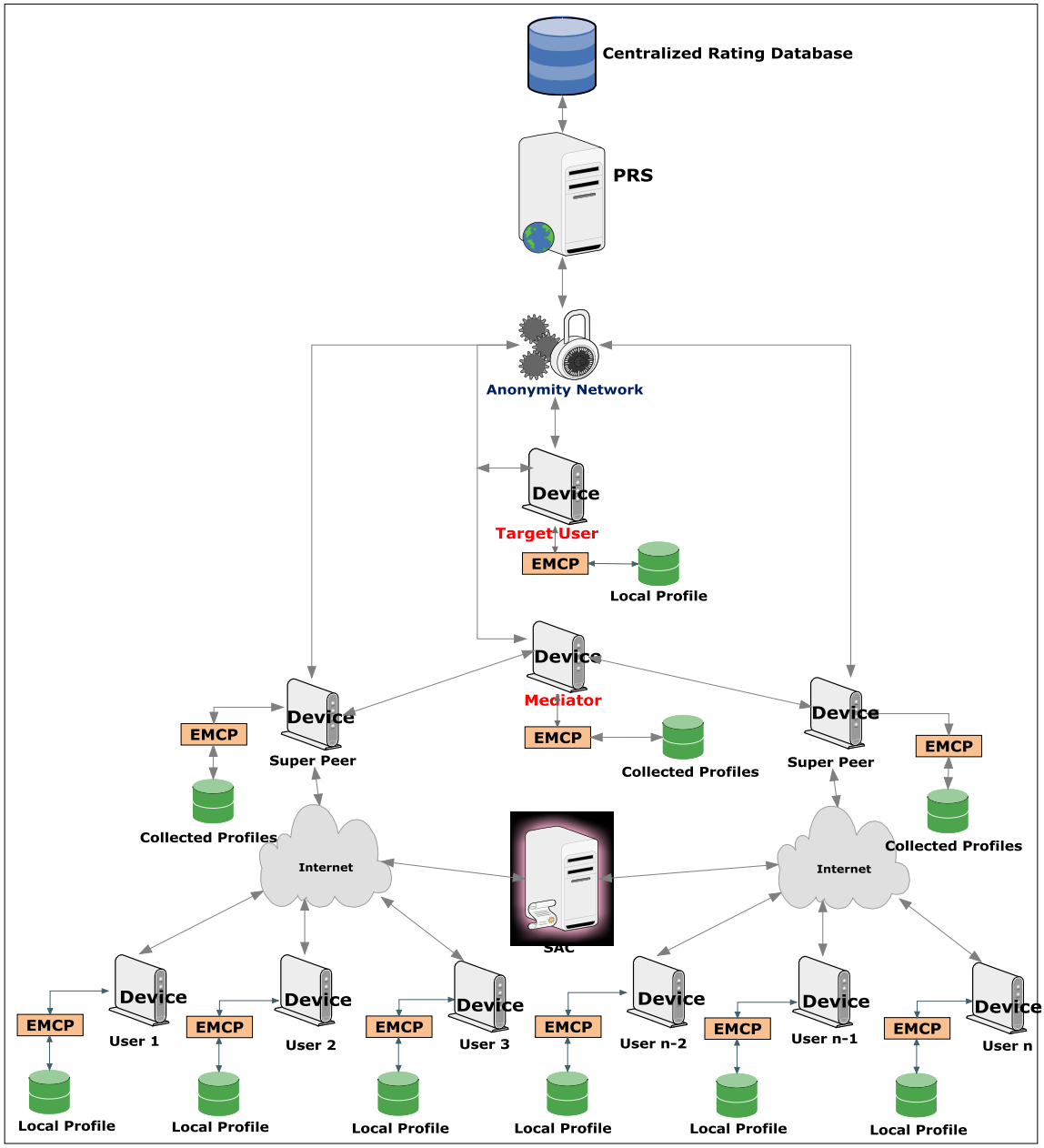


Figure 3.6: Overview of the Platform of Collaborative Privacy Framework

Moreover, homomorphic encryption possesses specific properties that permit computation of linear combinations of encrypted data without the need for prior decryption. Paillier [129] proposed a probabilistic asymmetric algorithm for public key cryptography that is an

example of an efficient additively homomorphic cryptosystem. This scheme is further extended by [130] with a threshold version, but required the use of a trusted dealer to distribute the keys to the participants. The reliance on a trusted dealer was lifted in [131] to ensure that no single party or coalition of less than the specific participants can recover the encrypted values. In designing the global concealment process, we require a fully distributed key generation protocol. In particular, the coalition between the recommender service (PRS) or target user with any super-peer within the peer-group should not be able to decrypt the whole aggregated profiles submitted to PRS. It only reveals the concealed profiles collected by this super-peer. Therefore, neither can be used as a trusted "dealer" for key generation. Thus, we employ a fully distributed threshold cryptosystem. Since it is desirable to distribute trust between numerous super-peers and no single super-peer is assumed to be fully trusted, then the decryption key sk is shared among a number P of super-peers, and encrypted profiles can only be decrypted if any subsets consisting of a threshold t of super-peers cooperate but no subset smaller than t can perform decryption. Moreover, with the additively homomorphic property of Paillier schema, it permits a secure aggregation and prediction over encrypted rating profiles. We assume a semi-honest model for the super-peers. Hence, we do not require zero-knowledge proofs (ZKPs) for the various cryptographic operations from the participants. We will briefly present the distributed Paillier threshold cryptosystem below.

Key Generation

In this step, each super-peer $\forall_{i=1}^{n} SP_i$ generates n additive shares of two $\kappa/2$-bit strong primes, such that each super-peer has share p_i and q_i. The method proposed in [131] is then used to compute $N = pq$, $\lambda = lCM\ (p-1, q-1)$, $g = N + 1$ such that $p = \sum_{i=1}^{n} p_i$, $\mathrm{q} = \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{q_i}$, also d such that $\mathrm{d} \equiv 1 \bmod \mathrm{N}$ and $\mathrm{d} \equiv 0 \bmod \lambda$. The public key $pk = (N, g)$ and the private key $sk = d$. Note that, super-peers perform the bi-primality

test in [132] for checking if N is a product of two primes in a distributed way. If the test fails, the protocol is restarted.

Key Sharing

The private key sk is shared among n super-peers with the Shamir scheme [130] as $t-1$ degree polynomial where each party obtain (t, n) share of d : Let $a_0 = d$, and randomly choose a_i in $\{0, \ldots N-1\}$ and set $f(X) = \sum_{i=0}^{t} a_i X^i$. The share s_i of the ith super-peer SP_i is $f(i) \bmod N$.

Encryption

To encrypt a message $M \in Z_N$ with public key , randomly choose $r \in Z_N^*$ and compute $C = g^M r^n \bmod N^2$.

Share Decryption

To decrypt C , each super-peer SP_i computes the decryption share $c_i = c^{2\Delta si} mod\ N^2$, where $\Delta = t!$ using his/her secret share s_i. Finally, if $t+1$ valid shares are available, they can be combined to recover M as described in End decryption.

End Decryption

Let S be a set of $t+1$ valid shares. Compute

$$M = L\left(\prod_{i \in S} c_i^{2\lambda_i} mod\ N^2\right)\frac{1}{4\Delta^2} mod\ N$$

Where $\lambda_i = \Delta \prod_{i' \epsilon S \setminus i} \frac{-i}{i-i'}$, See [131] for more details on the correctness of the scheme and for proofs of security.

The generated keys are stored in a database called “encryption key store”. More details about how referrals are generated and aggregating group profiles based on the distributed Paillier threshold cryptosystem were presented in Article VI and Article VII.

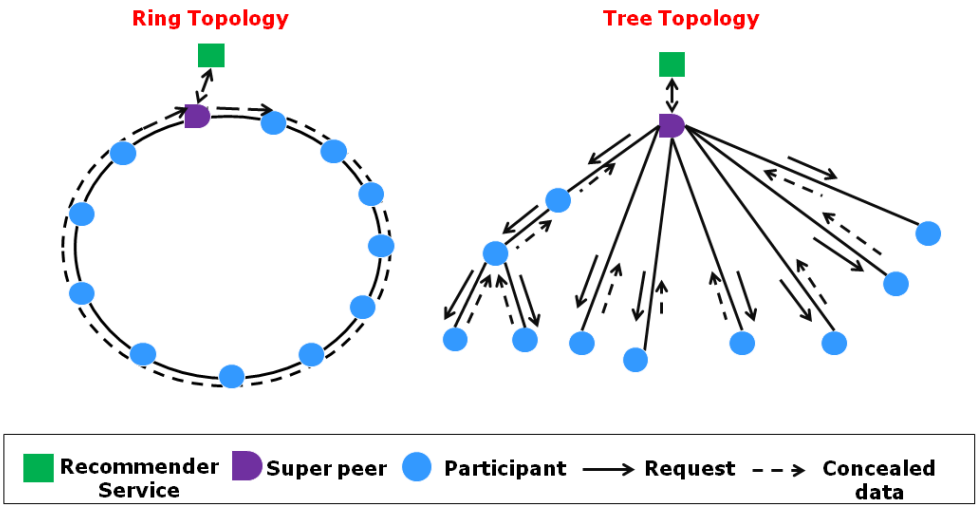


Figure 3.7: Topology for creating aggregated profile in peer-groups

Since the database is dynamic in nature, the local obfuscation agent periodically desensitizes the updated preferences, and then a synchronize agent forwards them to the private recommender service (PRS) or trusted peers upon owner permission. Thus, recommendation can be made on the most recent preferences. Moreover, the synchronize agent is responsible for calculating and storing parameterized paths in the anonymous network that attain high throughput, which in turn can be used in submitting preferences anonymously. These parameterized paths are stored in a database called "nodes store". The policy agent is an entity in *EMCP* that has the ability to encode privacy preferences and privacy policies as XML statements depending on the host role in the recommendation process. Hence, if the host's role is as a "super-peer", the policy agent will have the responsibility to encode the data collection and data usage practices as P3P policies via XML statements which are answering questions concerning the purpose of collection, the recipients of these profiles, and the retention policy. The P3P policies that are produced are recorded in a database called "policy store". Thereafter, each super-peer forwards these policies to the members of their peer-groups. On the other hand, if the host's role is as a "participant", the policy agent acquires the user's privacy preferences and expresses them using APPEL as a set of preferences rules which are then decoded into a set of elements that are stored in a database called "privacy preferences" in the form of tables called

"privacy meta-data". These rules contain both a privacy policy and an action to be taken for such a privacy policy. In this way, it will enable the preference checker to make self-acting decisions on objects that are encountered during the data collection process regarding different P3P policies (e.g., privacy preferences could include: certain categories of items should be excluded from data before submission, expiration of purchase history, usage of items that have been purchased with the business credit card and not with the private one, generalize certain terms or names in user's preferences according to defined taxonomy, using synonyms for certain terms or names in the user's preferences, suppressing certain items from the extracted preferences and inserting dummy items that have the same feature vector like the suppressed ones and limiting the potential output patterns from extracted preferences etc. , in order to prevent the disclosure of sensitive preferences in the user's profile). Query Rewriter rewrites the received request constrained by the privacy preference for its host. The security and privacy policies as well as their translation into the use of particular security and privacy mechanisms are outside the scope of this thesis. This is an interesting and important topic for future research. The OECD principles can potentially help in formulating these policies. An initial investigation of OECD role in our framework is discussed in section 3.2 of the thesis.

During the final phase, users access the recommended items. These generated referrals may be useful to enhance the service and future recommendations, since accessing those items does imply that the recommendation was correct. The feedback agent is responsible for anonymously submitting the participant's feedback/ billing information about the referred items and recommendations process to the super-peers of their peer-groups, which in turn send this information to the private recommender service (PRS) / content provider. Moreover, the feedback agent reports scores about the elected super-peer and the target-user to SAC. Finally, the recommender service returns a set of identifiers for items that might be interesting to the users. These identifiers are linked to items offered by their

content providers. The delivery agent is the entity which is responsible for communicating with the content provider in order to fetch the contents of the referred items. The recommendation process can be used to support the content distribution providers from different perspectives, such as maximizing the precision of target marketing and improving the overall performance of the current distribution network by building up an overlay to increase content availability, prioritization and distribution based on the referred items. Figure 3.9 shows the participants interactions with super-peers and the social recommender service. A general overview of the recommendation process in the proposed framework operates as follows:

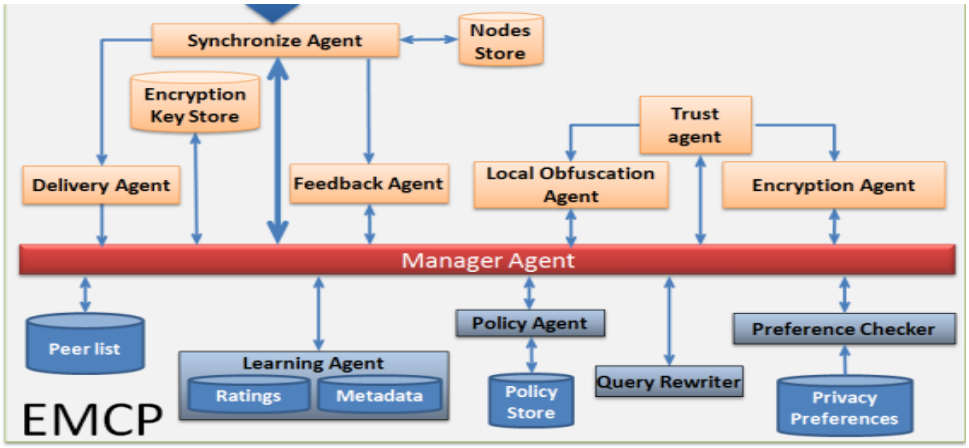


Figure 3.8: Inside *EMCP* Components

1. The target user (the user requesting recommendations) broadcasts a message to other users in the network requesting a recommendation for a specific genre or category of items. Thereafter, the target user selects a set of his preferences to be used later in the computation of the trust level at the participant side. The local obfuscation agent is employed to perform the local concealment process on the released data. Finally, the target user dispatches this data to the individual users who have decided to participate in the recommendation process.

2. Each group member negotiates with the security authority centre (SAC) to select a peer with the highest reputation to act as a "super-peer" which will act as a communication

gateway between the recommender service and the participants in its underlying peer-group. SAC is a trusted third party responsible for making an assessment on those super-peers according to the member' reports and super-peer reputations.

3. Each super-peer negotiates with both the target user and the recommender service to express its privacy policies for the data collection and usage process via P3P policies.

4. At the participant side, the manager agent receives the request from the target user along with the P3P policy from the elected super-peer. It then forwards the P3P policy to the preference checker and the request to the query rewriter. The preference checker ensures that the extracted preferences do not violate the privacy of its host which were previously declared by the use using APPEL preference. The query rewriter rewrites the received request based on the feedback of the preference checker. The modified request is directed to the learning agent to start collecting preferences that could satisfy the modified query and forwards it to the local obfuscation agent. Finally, the policy agent audits the original and modified requests plus estimated trust level and P3P policy with previous requests in order to prevent multiple requests that might extract sensitive preferences.

5. The trust agent calculates approximated interpersonal trust between its host and the target user in a decentralized fashion using the entropy definition proposed in [128] at each participant side. The entropy value becomes lower as the users' ratings are more consistent. For each two participants, $\forall_{j=1}^{n} T\left(u_a, u_{b_j}\right)$ is the estimated trust between the target user u_a and participant u_{b_j}. The whole process can be described using the following steps:

   i. Each participant $\forall_{j=1}^{n} u_{b_j}$ determines a subset of his/her preferences that will be required for the recommendation process. The participant then utilizes the shared preferences of u_a, u_{b_j} for the trust computation. Determining shared preferences is done by matching the received items' hash values from target user u_a with his/her local items' hash values.

ii. Participant u_{b_j} computes the trust level using equation

$$T\left(u_a, u_{b_j}\right) = \frac{Entropy(u_a) - Entropy\left(u_a \middle| u_{b_j}\right)}{Entropy(u_a)} \qquad (3.2)$$
$$= \frac{\left(1 - \frac{log\, N}{log\, ZN}\right) + \frac{1}{N\, log\, ZN}\left(\sum_{i=1}^{Z}\sum_{j=1}^{Z} n_{ij}\, log\, n_{ij} - \sum_{i=1}^{Z} n_i\, log\, n_i\right)}{1 - \frac{1}{N\, log\, ZN}\sum_{i=1}^{Z} n_i\, log\, n_i}$$

Equation (3.2) is an adapted formalization of trust as proposed in [128] where *Z* denotes the number of states of rated values and *N* is the total number of rating times. For example, if *Z=6* and *N=20* when 20 ratings are made with 1 to 6 integer valued scores. Employing entropy to select trustworthy neighbors achieves an improvement in the group formation and rating predication. The enhancement in rating predication is stemmed from trust propagation, so if $u_{b_{j=x}}$ is selected as a trustworthy user and he/she does not have a rating for the item to be predicted, a trustworthy user $u_{b_{j=y}}$ of user $u_{b_{j=x}}$ can also be used for the predication.

iii. Each participant $\forall_{j=1}^{n} u_{b_j}$ sends his/her calculated trust value to the super-peer. The estimated trust values are forwarded to both the super-peers and PRS.

iv. Each participant $\forall_{j=1}^{n} u_{b_j}$ sends this trust value to the local obfuscation agent to adjust the obfuscation level with the trust level. In other words, we correlate the obfuscation level with different levels of trust, so the more trusted a target user is, the less obfuscated copy of a users' preference he/she can access. The local obfuscation agent executes the local concealment process on items' ratings that are required in the recommendation process. Moreover, the local obfuscation agent hashes their identifiers and meta-data using LSH. The level of obfuscation is determined using the trust level with the target user.

v. Finally, the policy agent audits the original and modified requests plus estimated trust level and P3P policy with previous requests. This step allows *EMCP* to prevent multiple requests that might extract sensitive preferences. In such a case, if the target user requests same data twice, its trust level will be reduced, which will increase the

level of the obfuscation in the extracted preferences. This step will cause extracted preferences to appear as a completely different set of preferences to the target user.

The trust agent sends the calculated trust value to the super-peer. The estimated trust values are forwarded to both the super-peers and the recommender service. The concealed data is sent to the super-peers of the peer-group. Anonymous communication networks [133] can be utilized to hide the network identities of group members when submitting their concealed preferences to their super-peers.

6. Upon receiving the obfuscated preferences from the participants, each super-peer filters the received preferences based on the trust level of their owners such that $T\left(u_a, u_{b_j}\right) > \theta$ where θ is a minimum trust threshold value defined by the target user or PRS. Each super-peer then collects the participants' pseudonyms and builds a group profile (aggregated profile) such that all the <hashed value, rating> elements belonging to similar preferences are grouped together. This allows the computing of the preferences popularity curve at each super-peer. Each super-peer $\forall_{x=1}^{k} SP_x$ calculates the following intermediate values for each user in the N-neighbourhood of target user $\forall_{j=1}^{n} u_{b_j} \in \text{Neighbor}(u_a)$,

$$\text{Then } \forall\ q = 1 \ldots T \quad \widetilde{r_{u_{b_j},q}} = r_{u_{b_j},q} - \overline{r_q}$$

$$\widehat{r_{q,u_{b_j}}}^{x} = \frac{T\left(u_a,u_{b_j}\right) * \widetilde{r_{u_{b_j},q}}}{T\left(u_a,u_{b_j}\right)} \qquad (3.3)$$

Where $r_{u_{b_j},q}$ is the rating value of participant u_{b_j} for item q, $\overline{r_q}$ is the average rating for item q in each items' cluster. Then after each super-peer builds a group profile (aggregated profile), it performs the global concealment process in this profile. The super-peer can seamlessly interact with the recommender service (PRS) by posing as a user and has a group profile as his/her own profile.

7. The recommender service (PRS) runs the recommendation technique on this aggregated profile then forwards the referrals list along with their predicated ratings to the super-peer. Super-peers publish the final list to the target user and participants. Finally, each

participant reports scores about the elected super-peer of their peer-group and the target-user to SAC, which helps to determine the reputation of each entity involved in the referrals generation.

In order to demonstrate the applicability of this framework, this research focuses on five practical scenarios: IPTV recommender service, jukebox recommender service, data mash-up service, community recommender service, and location based recommender service. These scenarios are motivated by protecting user privacy while utilizing the service and its implications. The reason for selecting these scenarios was due to the fact that they represent the more pressing issues on privacy research and we hoped to enable the deployment of privacy-aware social recommender services using the collaborative privacy approach. Of course, other practical scenarios still exist for the proposed framework. In this thesis, we are unable to address all of them.

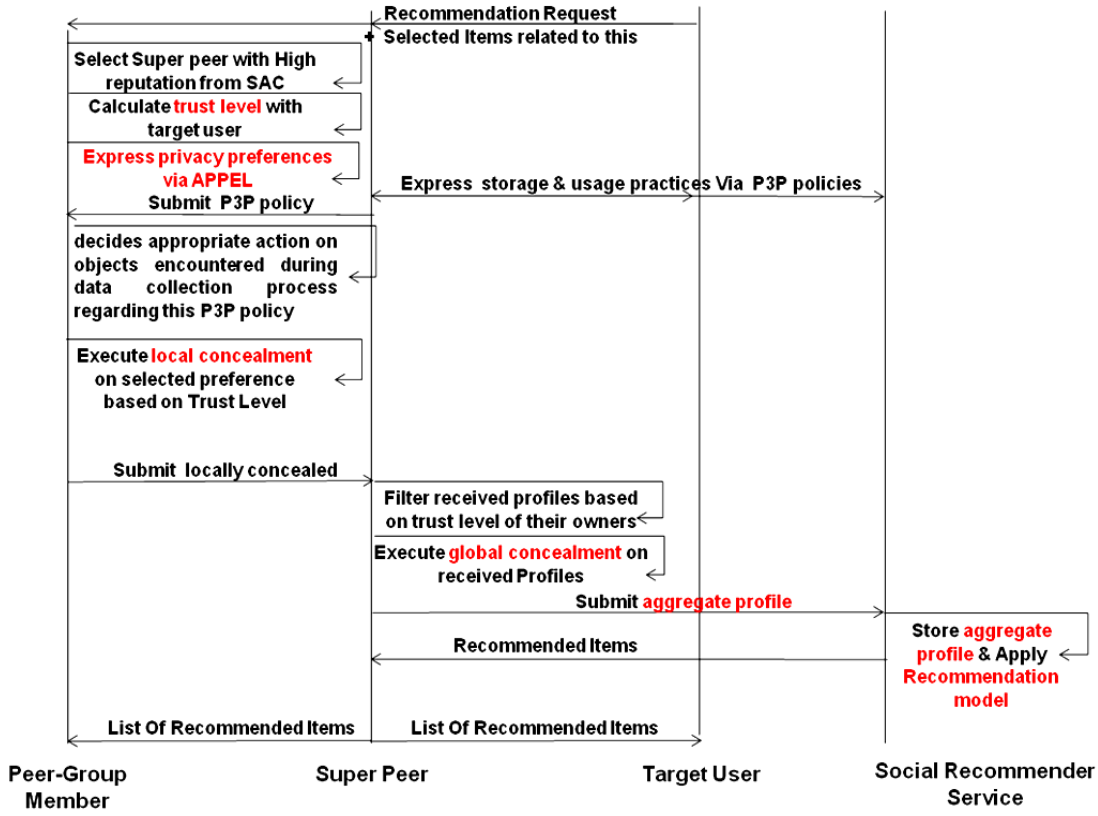


Figure 3.9: Interaction Sequence Diagram for the Collaborative privacy framework

Motivations and Restrictions of the Various Parties in Collaborative Privacy Approach

There are numerous motivations and restrictions for the various parties involved within our collaborative privacy framework, which make it not only valuable to the user but also to service providers. Our proposed middleware which is employed in the implementation of the framework permits the end-users to control the privacy of their released data while interacting with third-party social recommender services. This kind of approach is quite flexible and can easily be adopted in a conventional business model of the current service oriented based services, like social recommender services, because it is executed at the user side and it takes advantage of the social structure that is offered by the online content distribution service without the need for significant modifications at the service provider side. Moreover, service providers can also attain many benefits from adopting the proposed framework, such as, promoting a privacy friendly environment for their offered services, simplifying the data management process at their side and finally reducing their liability to secure their clients' personal information.

Motivations and Restrictions for Users

Users' Motivations

- Attaining ultimate control over their personal information: the users can determine for each recommendation request, what super-peers and purposes their data will be released for, and what data from their profiles gets collected in which concealment level. They are also aware of how long this data will be retained by external parties.
- Utilizing up-to-date data for recommendations purposes: storing the data locally at the user side facilitates the creation of accurate profiles and simplifies the update of these profiles with the most recent consumption history of these users. As a result, each time a recommendation request occurs, the users will release updated data from their current profile instead of using outdated data stored at the social recommender service, which

will allow the generation of accurate referrals that match their changing preferences and tastes.

- Specifying their privacy preferences: users can express their privacy preferences using APPEL as a set of rules which are then decoded into a set of elements that are stored in a privacy preferences database. These rules will enable *EMCP* to make self-acting decisions on data elements that are encountered during the data collection process regarding different P3P policies.
- Reducing the impact of privacy breaches: in case the occurrence of privacy invasion happens at the social recommender service, the leaked users' data will be worthless with a diminished informative value, because it is already concealed with a two stage concealment process and cannot be linked directly to a specific user within a peer-group. Moreover, the leaked users' data is concealed in a way to be only useful for recommendations purposes and it would be difficult to perform different kinds of analytical processes on such data.
- A third option for privacy aware users: privacy aware users will no longer have to choose between two options, either releasing their whole data to a recommender service which they have to trust or not using the service at all. Our collaborative privacy framework provides an alternative to the current models of practice.

Users' Restrictions

- The users have to formalize their privacy preference, which is a critical task, as the users need to realize various privacy concerns. They also need to deduce future recommendation requests that might raise privacy concerns for their collected data.
- The collaborative privacy framework does not fully protect users from malicious super-peers. The malicious super-peer can uncover the user's anonymity during the release of his/her data to a specific recommendation request. This problem has been mitigated by

utilizing anonymity networks while sending the data from users to the super-peer and employing reputation mechanisms in order to select proper super-peers with a stable success rate. Moreover, the user's data is not in a raw form and its privacy is already protected with a local concealment process before leaving the user's device.

- Within the proposed collaborative privacy framework, the user's profile is stored at his/her local machine in a raw form, which makes it vulnerable to malware/spyware that might infect this machine. In order to mitigate this problem, the user's profile is encrypted with a secret key encryption algorithm when the user is not using the system. Moreover, special considerations need to be added to the operating system in order to ensure strong safety and trustworthy guarantees to the middleware in the running memory and storage of users' machine even in the presence of a malicious software (sandboxing, intrusion detection, antivirus …etc.)

Motivations and Restrictions for Recommender Service Providers

Service Providers' Motivations

- Providing accurate referrals: the referrals are extracted from up-to-date data, which is collected prior to the start of the recommendation process. This has a number of beneficial advantages for the offered service, such as, reducing the users' frustration, increasing the number of potential users for the service, and raising the revenue of the service providers.
- Using the current social recommendation techniques: adopting the collaborative privacy framework does not require the design of new recommendation techniques, the current off-the-shelf social recommendation algorithms can be used directly on the concealed data without the need to return it back into a raw form.
- Readiness to be used in the conventional business model of the current service oriented based services: most of the existing service providers find difficulties in integrating

privacy enhancing technologies within their service, as the addition of privacy and cryptography components requires a significant change on their service's back- end infrastructure. Our collaborative privacy framework utilizes the user and social sides of the service providers as an infrastructure for the implementation of our framework. The collaborative privacy framework is quite flexible and can easily be adopted in the current business model of social recommender services because it is executed at the user side and takes advantage of the social structure that is offered by their service without the need for significant modifications at the service provider side.

- Simplifying the data management process at the service side: within the collaborative privacy framework, the users' profiles are stored on their side on their own devices. However, in order to enable the service providers to use the users' data in more sophisticated business processes, a concealed public version of users' profiles are stored on their side to serve the enterprise business' initiatives of these service providers.
- Promoting a privacy friendly environment for the offered referrals: Privacy aware users will be encouraged to participate in such a service, since their personal data will be stored locally on their own side and they can decide what data to be released for every request. In addition, the released data will not leave their devices until it is properly concealed.
- Reducing the liability of service providers in securing their clients' personal information: the responsibility of the service providers for protecting their clients' personal data is alleviated, as the clear and accurate version of users' profiles are stored on the users' devices. Privacy invasion on these public profiles will not be as harmful as much as it is when compared with the ones that occur in the current conventional approaches of privacy.
- Enhance the efficiency of the content distribution providers: the extracted recommendations can be used to support the content distribution providers from

different perspectives, such as maximizing the precision of target marketing and improving the overall performance of the current distribution network by building up an overlay to increase content availability, prioritization and distribution based on the predicated recommendations.

Service Providers' Restrictions

- Losing the control over users' profiles: indeed, the users' profiles are stored remotely at their side. However, the service providers are also holding and storing public profiles from previous recommendation processes. Although the public profiles are an outdated snapshot of the users' data in a concealed form, they are sufficient enough for training, building, and maintaining the recommendation model.
- Potential abuse of the service by malicious users: the anonymity attained by our collaborative privacy approach can induce malicious users to perform attacks on the service or other users while exploiting the advantage of hiding their identity, thus they can escape from legal prosecution. We have introduced the usage of a security authority centre (SAC), which is a trusted third party responsible for assessing the reputation of each entity involved in the referrals generation process. Moreover, SAC is in charge of issuing anonymous credentials for each user in the system. Future research should investigate how to attain the functionality of SAC in P2P fashion and without relying on a centralized entity.

Privacy Enforcement

Utilizing topological formation for data collection with a two stage concealment process within our framework allows the user to control what data from their profiles gets collected and in which concealment level. Specifically, the public group profile that is exposed to the third party social recommender services contains a set of collected items from the users' profiles that are released to a specific recommendation request. These items usually

represent a small proportion of items in relative relation to the total number of consumed items in the users' profiles. Moreover, the anonymity and concealment techniques used during the data collection process ensure attaining an appropriate privacy level for system users. Those are very important aspects in our framework that depict its ability to diminish the impact of the privacy breaches, limit the misuse of personal information, and to enforce and verify the attained privacy for its users. Moreover, using P3P policies enable the user to present evidence that his/her preferences were released for a specific recommendation process, at a specific time, and for a specific super-peer.

A set of stochastic techniques based on the machine learning clustering analysis were proposed in Articles III, IV, V, VI, VII and VIII (for the recommender service for IPTV content providers scenario), Articles IX and X (for the jukebox content recommender service scenario), Articles XI, XII and XIII (for the community discovery & recommendation service), and finally Article XVIII (for the pervasive healthcare service scenario). In order to quantify the achieved privacy level and accuracy of referrals, as a rule of thumb, a higher level of privacy leads to lower accuracy. We have selected generic metrics which were partially presented in each of the previous articles.

3.4 Privacy Aware Data Mash-ups For IPTV Recommender Service

In this section, an overview of the scenario was presented where the data mash-up service (DMS) integrates datasets from multiple online movie database sites for a recommender service running at the IPTV provider. The data mash-up process based on *EMCP* can be summarized as follows: the recommender service sends a query to the DMS to gather information for some genres to leverage its recommendation model. The DMS lookup in its providers' cache to determine the providers that could satisfy that query then it transforms the recommender service's query into appropriate sub-queries language suitable for each provider's database. DMS sends each sub-query to the candidate providers to incite them

about the data mash-up process. The provider who decides to participate in that process forwards the sub-query to its local *EMCP* that rewrites it considering its privacy preferences. *EMCP* extracts the dataset satisfying this modified sub-query and then performs a local concealment process on this dataset to hide the subscribers' preferences. After receiving all the locally concealed data from all the online movie database providers, DMS builds a virtualized schema from all datasets then executes a global concealment process on the aggregated datasets. Finally, DMS deliver the resulting datasets to the recommender service. We used anonymous pseudonym identities to alleviate the providers' identity problems, as the database providers do not want to reveal their ownership of the data to the other competing providers. Moreover, the DMS will be keen to hide the identities of its clients as a business asset.

Evaluation Results

In order to evaluate the two stage concealment process proposed in this scenario, the mean average error (MAE) metric proposed in [134] was used to measure the accuracy of generated recommendations and mutual information metric was used to measure the privacy breach level. Regarding the local concealment algorithm proposed in this scenario, in order to measure the relation between the quantity of real items in the locally concealed dataset and privacy breach, α values were selected in range from 1.0 to 5.5, and then the number of real items was increased from 100 to 1000. The fake-item set was selected using uniform distribution as a baseline. As shown in Figure 3.10, the locally concealed fake set reduces the privacy breach and performs much better than the uniform fake set. As the number of real items increase, the uniform fake set gets worse as more information is leaked while the generated concealed fake set is not affected with that attitude. The obtained results are promising especially when dealing with a large number of real items.

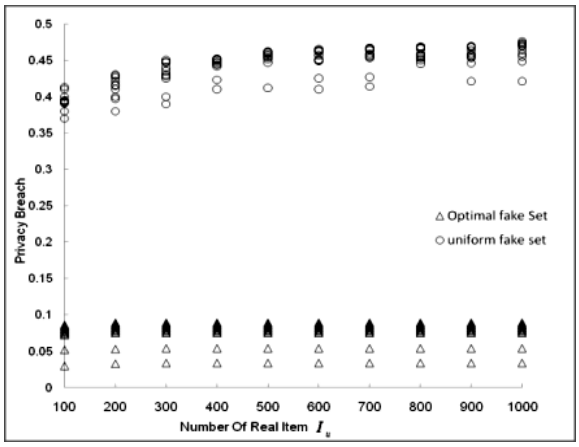


Figure 3.10: Privacy breach for optimal and uniform fake sets

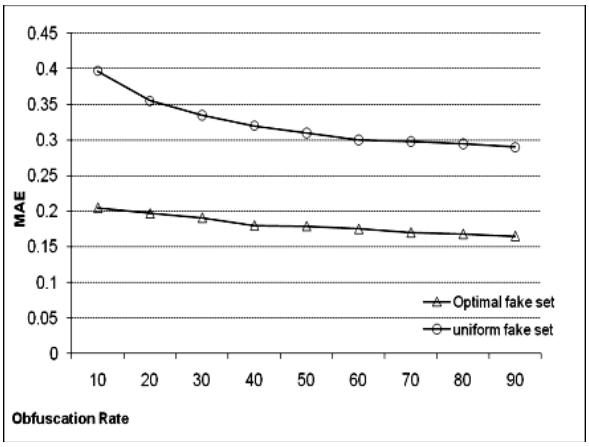


Figure 3.11: MAE of the generated predications vs. obfuscation rate

In the next experiment, the relationship between the quantity of fake items and the accuracy of recommendations was measured. The percentage of real items in D_{ϖ} was gradually increased from 0.1 to 0.9. Thereafter, for each possible obfuscation rate, MAE values for the whole concealed dataset were measured. Figure 3.11 shows MAE values as a function of the obfuscation rate. The provider selects the obfuscation rate based on the desired accuracy level required from the recommendation process, such that, with a higher value for the obfuscation rate, the higher the accuracy of the recommendation that the user can attain. Adding items from the optimal fake set has a minor impact on the MAE of the generated recommendations without having to select a higher value for the obfuscation rate. However, as seen from the graph, the MAE rate slightly decreases in a roughly linear manner with higher values of the obfuscation rate. In particular, the change in MAE is minor in the range of 40% to 60% which confirms the assumption that accurate recommendations can be provided with lower values for the obfuscation rate. The optimal fake items are so similar to the real items in the dataset that the obfuscation does not significantly change the aggregates in the real dataset and it has a small impact on MAE.

Regarding the global concealment algorithm, the effect of ρ values on the accuracy and privacy for the overall recommendations was evaluated. ρ values were varied from 0 to 100

to show how different values affect accuracy and privacy. Note that when ρ is 0, this means selecting all unrated items and filling them with random values chosen using a distribution reflecting the ratings in the merged datasets.

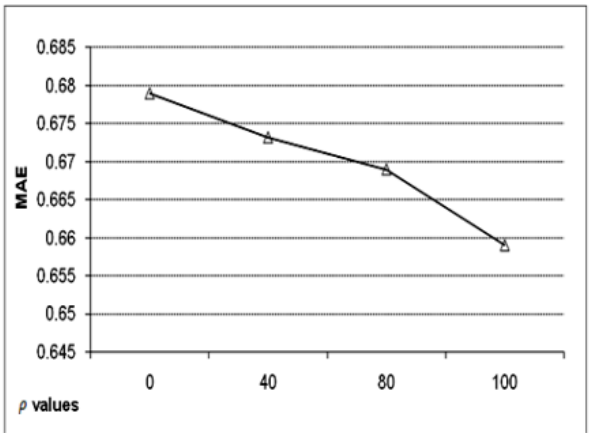


Figure 3.12: MAE of the recommendations for different ρ values

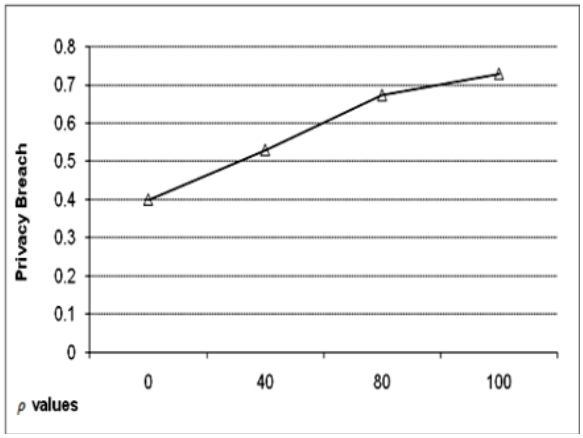


Figure 3.13: Privacy breach of the concealed Data for different ρ values

The calculated values of MAE and privacy breach are shown in Figures 3.12 and 3.13. As seen from Figure 3.12, the accuracy becomes better with augmented ρ values, as the size of the selected portion that filled using *KNN* increases and the size of the randomized portion decreased. Although augmenting ρ values attains lower MAE values, we still have a decent accuracy level for recommendations. Accuracy losses result from errors in predications such that the predicted ratings might not represent the true ratings for these unrated items. There is also an error yield from using *KNN* predications with different values for *K*. These errors can guarantee a lower limit for privacy breach for the merged datasets as shown in Figure 3.13. This contributes to overcoming some privacy breaches that might happen due to data mash-up from multiple independent sites [135]. Finally, we can easily conclude that the accuracy losses due to privacy concerns are small and the proposed global concealment algorithm makes it possible to offer accurate recommendations.

3.5 Privacy Aware Recommender Service for IPTV Content Providers

In this section, an overview of the scenario was presented where a private centralized recommender service (PRS) is implemented as an external third party service where IPTV content providers deliver their users' preferences in order to receive recommendations. The interaction sequence in our collaborative privacy framework relies on super-peers which are trusted aggregators to produce aggregated group profiles and execute a global concealment process. Super-peers are selected based on their reputation reported at a security authority centre (SAC). However, in some cases of this scenario, we assumed that SAC is absent. As a result, we have employed the target user as a trusted aggregator for its peer-group. We also alleviate the users' identity problems by using anonymous pseudonym identities for users. The recommendation process based on the two stage concealment process in our framework can be summarized as following:

1. The target user broadcasts a message to other users in the IPTV network to start the recommendations process. Moreover, he/she runs the local concealment process on his /her profile.
2. Individual users that decide to respond to that request from peer-groups then elect a super-peer within each peer-group. Each participant within the peer-group executes the local concealment process on his/her profile's preferences. Then afterward, participants submit their locally concealed profiles' preferences either to the requester or to the super-peer (trusted aggregator) of their peer-group.
3. The trusted aggregator aggregates the collected preferences then executes a global concealment process on the aggregated group profile.
4. The trusted aggregator submits the globally concealed group profile together with pseudonyms of users who participate in the recommendation process to PRS.

5. The PRS inserts the received data in its database, updates its model using the received group profile and then it extracts the recommendation list for this group profile. Finally, PRS submits this list to the trusted aggregator.
6. The trusted aggregator distributes this list to the other members of peer-groups in order to attain a good reputation between them.

Evaluation Results

In order to evaluate the two stage concealment process proposed in this scenario, the mean average error (MAE) metric proposed in [134] was used to measure the accuracy of generated recommendations and the variation of information (VI) metric was used to measure the privacy breach level. Regarding the local concealment algorithm proposed in this scenario, the impact of the varying portion size and number of core-points on variation of information (VI) of the transformed ratings were measured. At the start, the portion size has been kept constant with a different number of core-points and then the portion size has varied with a constant number of core-points. Based on the results shown in Figures 3.14 and 3.15, the following remark can be deduced, when the number of core-points is small, the VI values are high but these values slowly decrease when increasing the number of core-points. At a certain point, VI values rise to a local maxima then decrease. Finally, VI values rise again with the growing number of core-points. It can be justified that VI values are high with a lower number of core-points as any point can move from one core-point to another. Moreover, with a sufficient number of core-points, there is a little chance of a point to move from one core-point to another, which causes the increase in VI values. Each user in the network can control his/her privacy by diverging different parameters for the local concealment algorithm.

Regarding the global concealment algorithm proposed in this scenario, the impact of sample size on the accuracy level was measured.

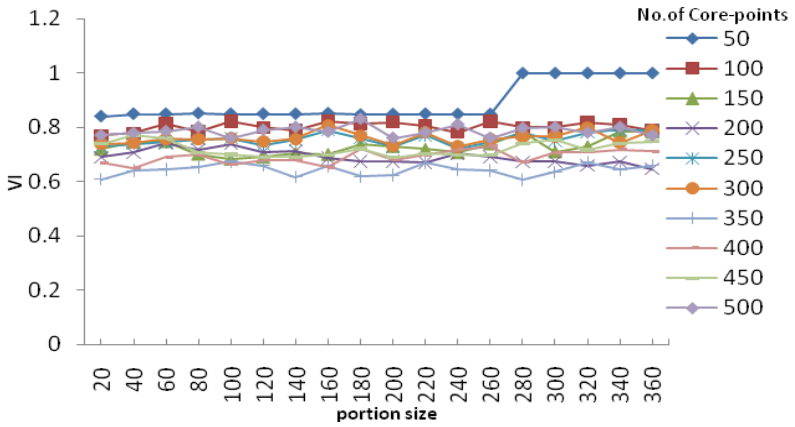


Figure 3.14: VI for different portion size

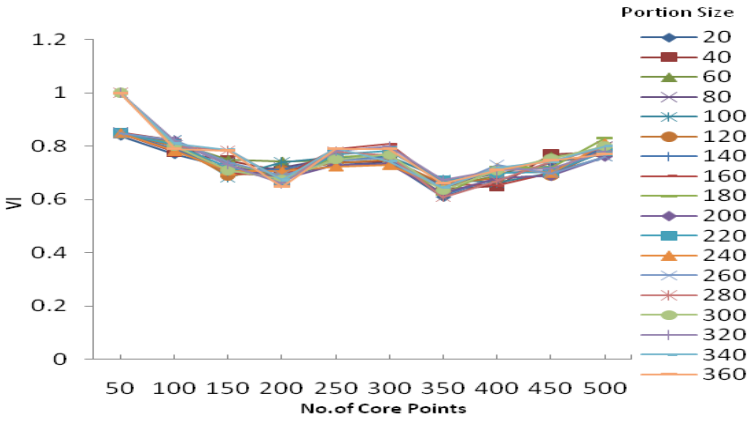


Figure 3.15: VI for different number of core points

A specific threshold value for the minimum number of responding users for each recommendation request was fixed to be 100 users. As shown in Figures 3.16, the increase in sample size leads to higher accuracy in the generated recommendations. However, at a certain sample size, the accuracy of the recommendations starts to decrease again due to the data loss in the sampling process.

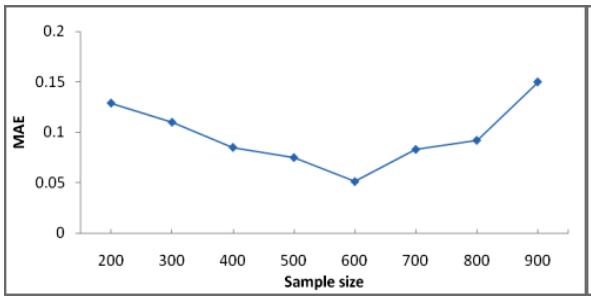


Figure 3.16: Relationship between sample size and MAE

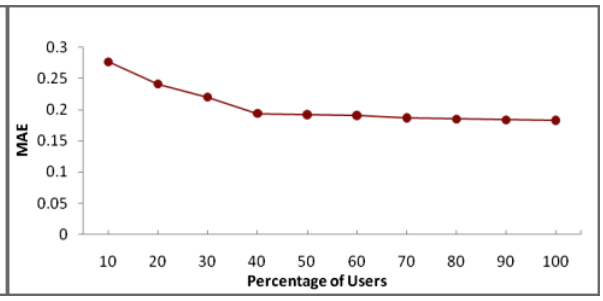


Figure 3.17: Relationship between percentage of Users and MAE

Moreover, the impact of changing the number of users involved within a certain request on the accuracy of the recommendations was measured. A general case was simulated where the number of users was fixed to be 50.000. Different numbers of users were assigned to a certain recommendation request, and then the percentage of users who joined this request was gradually increased from 10% to 100% of them. The parameters of the two stage concealment process were fixed and the MAE's values for the generated results were measured. As shown in Figure 3.17, the MAE value that occurs at approximately 40% of the users is close to the MAE value for all users. The main conclusion is that, with a low percentage of users, the MAE value is close to the original MAE value for all users. As a result, the target user does not need to broadcast the request to the full network to attain accurate results but he/she can employ multicast for certain users stored in his/her peer list to reduce the load in network traffic.

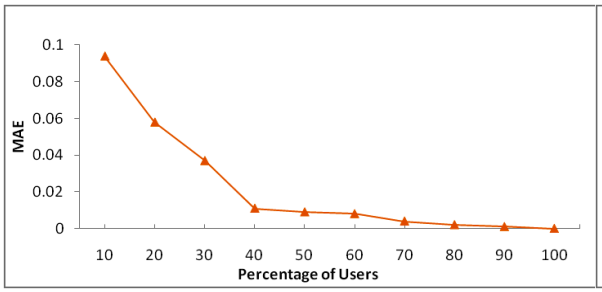


Figure 3.18: Relation between MAE values and percentage of users

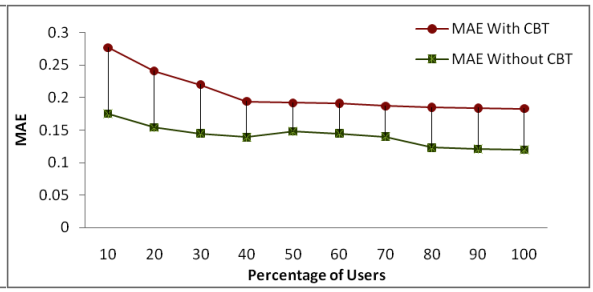


Figure 3.19: Influence of applying the local concealment stage

To illustrate the decrement of MAE values for recommendations based on diverse percentages of users groups and the whole users in the network, Figure 3.18 was plotted. This verifies the conclusion that the MAE value approximately converges to the MAE value which is obtained using the whole users in our case. The final experiment was conducted to measure the impact of using the local concealment algorithm as a pre-processing step for the global concealment algorithm. As presented in Figure 3.19, using the local concealment algorithm increases MAE values for lower percentages of users compared to using the global concealment algorithm alone. This can be explained, since the

distortion effect of the local concealment algorithm will be clearly visible for a lower percentage of users. However, augmenting the percentage of users who participate in a certain request has an effect on scaling down the error in MAE values. Finally, the results presented in these experiments show that the resulting recommendations obtained from the dataset pre-processed with our two stage concealment process is quite similar in the accuracy to the ones generated from the original dataset. The results also clarify that the proposed algorithms preserve the utility of the data to some degree such that reasonable recommendations can be obtained without enforcing the target user to collect profiles from numerous users. As a result, only a small percentage from users is needed to attain that goal. The various approaches which were taken for attaining users' profiles privacy along with the set of PETs to achieve this goal are described in more detail in Articles III, IV, V, VI, VII and VIII.

3.6 Private Community Discovery & Recommendation Service

In this section, an overview of the scenario was presented about the community discovery and recommendation service, and the issues related to the privacy of the users' profiles in the community building process were also analysed. Close inter-user interactions is the key privacy challenge in the community building process due to the diversity and massive size of user generated profiles. The scenario that we are targeting can be summarized as follows: an administrator or manager can employ a community based recommender service in order to facilitate social and professional interaction between various users from different backgrounds. The produced groups (sub-communities) tend to evolve out of collaborating members with similar preferences and the participation of new members. Various recommendations can be obtained when running the recommender service on these profiles while respecting privacy constraints and requirements of their owners by enabling them to have control over what parts of their profiles they are willing to share and in which

granularity. The architecture of the community recommender service in a university is depicted in Figure 3.20. The community recommender service (CRS) can be utilized to provide students with personalized referrals for joining specific sub–communities that are similar to their preferences. In this architecture, a profile is used to capture the updated preferences of each user. It typically includes demographic information besides other preferences for joining various sub–communities. Each extracted community consists of various sub-communities' profiles where each sub-community's profile keeps track of all the data related to this sub-community. This data represents the collective concealed preferences for different members within this sub-community.

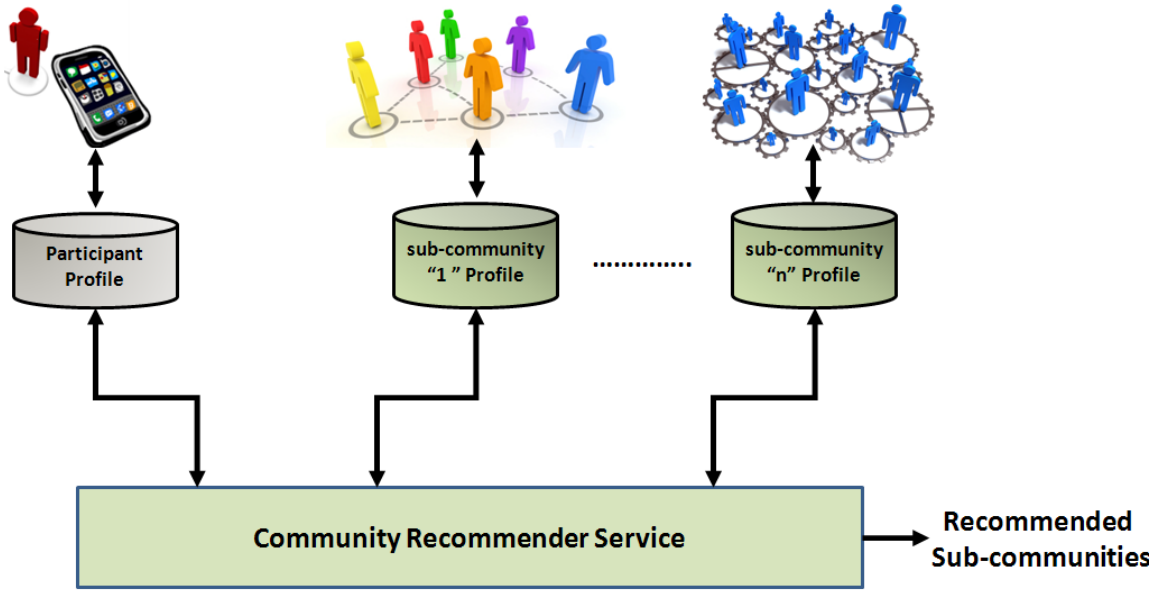


Figure 3.20: Generating Recommendations for Participants

Employing our collaborative privacy approach allows the users to release their profiles in a concealed form to super-peers. These super-peers then collaborate together in categorizing the produced communities into various sub-communities profiles. These final sub-communities' profiles assist CRS to offer referrals to new members based on the similarity between their profiles and these sub-communities profiles. Assigning a new student to a sub-community could implicitly update the formulated sub-communities profiles. In that case, a new student profile does not have enough preferences for generating referrals;

recommendations can be made using his/her demographics information after locally concealing it. The interaction sequence can be summarized as follows: based on various topics and activities in the university/conference, administrators can propose different communities of which each has its own interaction space where any interactions are supported. Each participant configures his/her *EMCP* to build a concealed public profile using the local concealment process. Peer-groups are formed with different participants along with super-peers for collecting concealed profiles from participants in each peer-group. Super-peers aggregate locally concealed preferences into a group profile and then extract different communities from this aggregated group-profile. Participants within each community encrypt their private profiles then engage in peer to peer communication between one another to discover different sub-communities within the community they belong to. Super-peers receive the discovered sub-communities' profiles and then send the representatives of each sub-community to both CRS and community members.

Evaluation Results

In order to evaluate the two stage concealment process proposed in this scenario, the famous precision and recall metrics were used to measure privacy and accuracy of the results. The accuracy precision measures the portion of interests in a specific sub-community that certain user likes, while recall measures the portion of interests possessed by each user which are actually in the joined sub-community. However, to measure the privacy or distortion achieved using the proposed protocols, the previous metrics were used to measure the true positive interests that are inferred from user's private profile when he/she joins a specific sub-community, as these interests might be shared between all sub-community members. As a result, precision will measure the portion of interests that are shared by members and that are true private interests for the user. Recall refers to the portion of private interests possessed by this user that are actually in these shared interests (privacy leak).

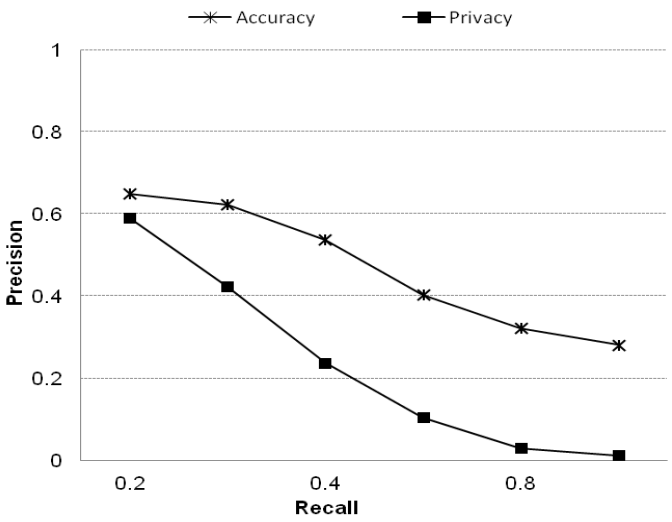


Figure 3.21: Recommendations accuracy and privacy

Regarding the evaluation of recommendations' accuracy, the results were shown in Figure 3.21. As seen in this figure, good quality is achieved due to creating generalized communities in the start that involve various groups which enable highly selective recommendations for the users in this community, since each community gathers participants who share the same general interests. Moreover, the effect of each interest inside every community can be easily measured, which in turn enables the detection and removal of outlier interests that are different than the general interests.

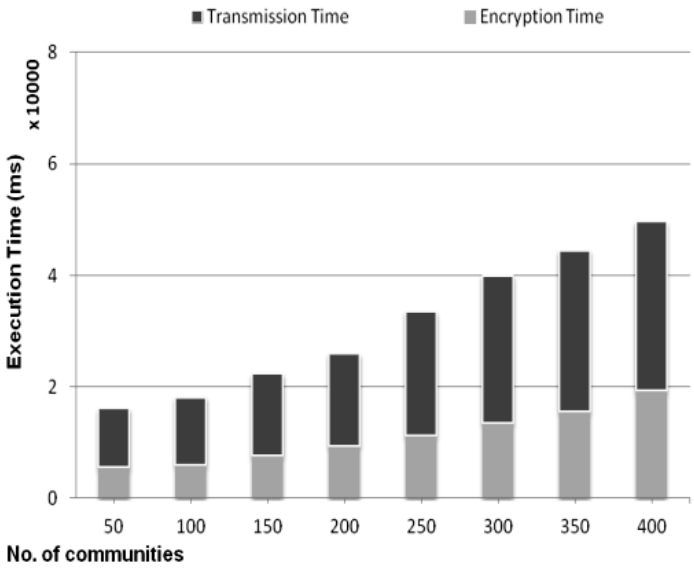


Figure 3.22: Efficiency of our solution

In the second experiment, the leaked private interests of different users were quantified. Users who published a portion of their real interests in their public profiles were considered, where for each of these users, the hidden interests in their profiles were inferred based on the community that they belong to. The interests obtained were quantified using the proposed metrics and the results are shown in Figure 3.21. As seen, the two stage concealment process managed to reduce privacy leakages for the exposed users' private interests. One important notice to put in consideration is that privacy metrics are pessimistic as the disclosed hashed interests agreed and published by the users in their public profiles other interests are hashed hypernym terms for their private interests. The private profile is hashed and encrypted during the computation. Moreover, sub-community joining is determined at the user side. Therefore, such information disclosure has a limited impact on the private interests' breach. On the other hand, sub-communities are represented with two values and collected users' profiles are omitted from submission to the community recommender service. In the next experiment, the efficiency of the proposed solution with an increasing number of communities was measured. The execution time was measured in terms of encryption and transmission time for users' profiles, as seen from Figure 3.22. The proposed solution requires more communication due to the distributed design and communication needs for the two stage concealment process. This acceptable overhead is shared among all users while the benefit is to protect their privacy without hampering the recommendation quality.

Regarding the local concealment protocol, its execution time was evaluated. One user got 60% of the total number of records and the rest of the records were divided to other clients as parts of approximately same number of records. In the second one, one client got 40% of total number of records while the other clients got the rest. The result of this experiment was summarized in Figure 3.23. This result indicates the performance benefits of the local concealment protocol, since it is not sensitive to the number of shared interests.

Regarding the global concealment protocol, the accuracy of the extracted sub-communities was measured. In order to compare the accuracy of the produced results, hierarchical agglomerative clustering was applied on the dataset in order to identify the natural sub-communities from the users' private profiles. These sub-communities are utilized for measuring the accuracy of the results produced by the global concealment protocol. Each cluster represents a sub-community which is constructed from a set of users' private profiles who share the same specific interests about the same topic. To measure the quality of the results, the two error metrics defined in [136] were used, which are grouping error (CR) and critical error (CIE). The first one, the grouping error (CR), takes into account the number of users' profiles included in a specific sub-community, but belonging to a topic different from the dominant topic in that sub-community. The second one, the critical error (CIR) measures the number of attendees' profiles belonging to a specific topic that is not the dominant one in any sub-community. The graphs in Figures 3.24 and 3.25 demonstrated both the CR and CIE values for the results obtained from the hierarchical clustering and global concealment protocol for a different number of sub-communities. This experiment is performed on two versions of the dataset; users' generalized profiles are utilized by the global concealment protocol, while hierarchical agglomerative clustering utilizes users' private profiles that should be kept private at the user side. Both CR and CIE for the global concealment protocol decrease with the increase in the number of sub-communities until reaching the natural number of sub-communities. This indicates that achieving privacy is feasible and does not severely affect the accuracy of the generated sub-communities.

In the last experiment on the global concealment protocol, the overhead of the execution time was measured when applying the global concealment protocol to preserve users' privacy. The dataset was divided into different numbers of records from 30.000 to 67.000, such that each party held approximately the same number of records.

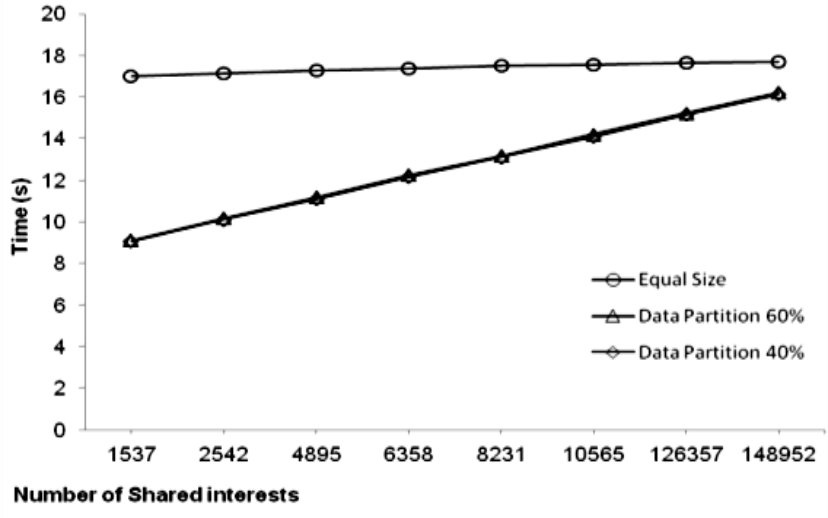


Figure 3.23: The Execution Time for Local Concealment Protocol

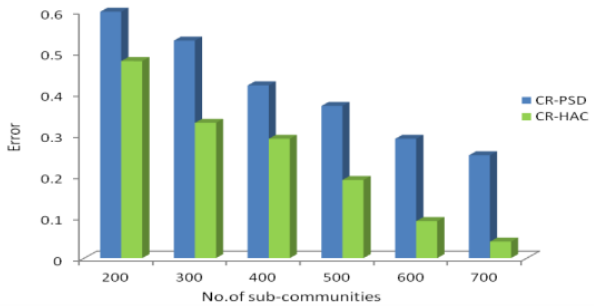


Figure 3.24: Grouping Error (CR) of the Global Concealment Protocol

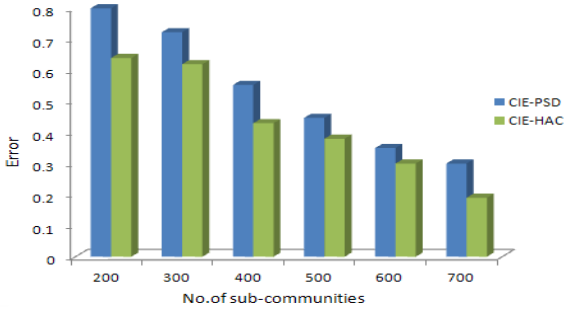


Figure 3.25: Critical Error (CIR) of the Global Concealment

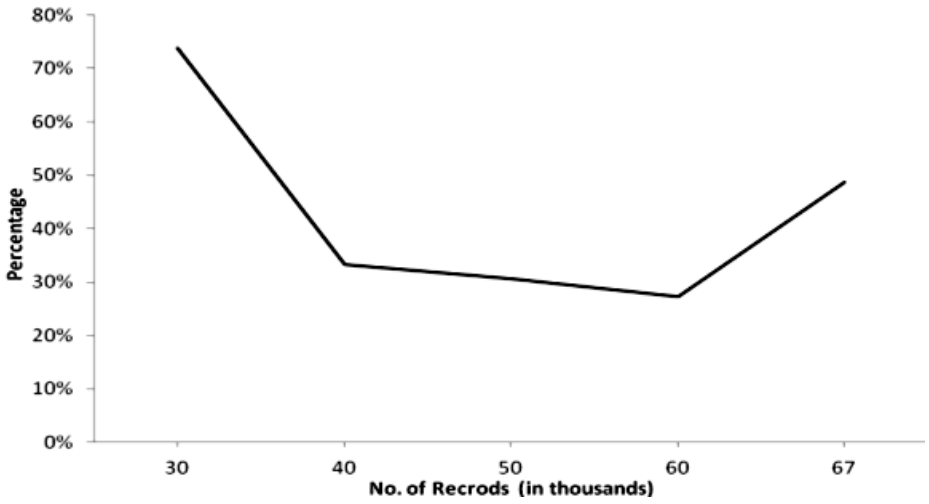


Figure 3.26: Percentage Time for the Global Concealment Protocol on Different Number of Records

The execution time was recorded when applying the global concealment protocol with encryption and without encryption on this data, and then the percentage was calculated as following:

$$percentage = \left({time\ without\ encryption}/{time\ with\ encryption} \right) * 100\ .$$

The graph in Figure 3.26 shows a time comparison of the global concealment protocol with and without encryption for different sizes of datasets. From the results, the proposed global concealment protocol has a reasonable performance and the privacy preserving nature has a marginal impact on the execution time in comparison with the non-encryption option. The various approaches which were taken for attaining users' profiles privacy along with the set of PETs to achieve this goal are described in more detail in Articles XI, XII, and XIII.

3.7 Privacy Aware Mobile Jukebox Recommender Service

In this section, an overview of the scenario was presented where a social recommender service (PRS) is implemented on an external third party server and end-users give information about their preferences to that server in order to receive music recommendations. The user preferences are stored in his/her profile in the form of ratings or votes for different items, such that items are rated explicitly or implicitly on a scale from 1 to 5. An item with a rating of 1 indicates that the user dislikes it while a rating of 5 means that the user likes it. The recommender service collects and stores different users' preferences in order to generate useful recommendations. There are two possible ways for the user's disclosure: through his/her personal preferences included in his/her profile [137] or through the user's network address (IP). *EMCP* employs two principles to eliminate these two disclosure channels, respectively. The two stage concealment process was used to conceal user's preferences for different items in his/her profile and an anonymous data

collection protocol is used to hide the user's network identity by routing the communication with other participants through relaying nodes in Tor's anonymous network [138]. In this scenario, the mobile phone storage is used to store the user profile. However, the mobile jukebox recommender service maintains a centralized rating database for storing the group profiles that are used in model building. Additionally, we alleviate the user's identity problems stated above by using anonymous pseudonyms identities for users. The recommendation process based on the two stage concealment process in our framework can be summarized as follows:

1. The learning agent collects user's preferences about different items which represent a local profile. The local profile is stored in two databases, the first one is the rating database that contains (id, rating) and the other one is the metadata database that contains the feature vector for each item (id, feature1, feature2, feature3). The feature vector can include: genre, author, album, decade, vocalness, singer, instruments, number of reproductions, and so on.
2. The target user broadcasts a message to other users near him/her to request recommendations for a specific genre or category of items. Individual users who decide to respond to that request perform the local concealment process to conceal a part of their local profiles that match the query. The group members submit their locally concealed profiles to the requester using an anonymized network like TOR to hide their network identities.
3. After the target user receives all the participants' profiles (group profile), he/she executes a global concealment process to conceal the group profile. Then he/she can interact with the recommender service by acting as an end-user and have the group profile as his/her own profile. The target user submits the group profile through an anonymized network to the mobile jukebox recommender service in order to attain recommendations.

4. The mobile jukebox recommender service performs its filtering techniques on the group profile which in turn returns a list of items that are correlated with that profile. This list is encrypted with a private key provided by the target-user and it is sent back on the reverse path to the target user that in turn gets decrypted and published anonymously to the other users that participated in the recommendation process.

Evaluation Results

To evaluate the accuracy of the local concealment algorithm proposed in this scenario with respect to a different number of dimensions in the user profile, the *d-dim* parameters of local concealment algorithm was varied to control the number of dimensions during the local concealment process. Figure 3.27 shows the performance of recommendations of locally concealed data in terms of mean absolute error (MAE). It is shown that the accuracy of recommendations based on the concealed data is slightly low when *d-dim* is low. But at a certain number of dimensions (500), the accuracy of recommendations on the concealed data is nearly equal to the accuracy obtained using the original data. In the second experiment performed on the local concealment algorithm, the effect of the *d-dim* on privacy level attained in terms of the variation of information (VI) metric was examined. As shown in Figure 3.28, privacy levels decrease with respect to the increase in *d-dim* values in the user profile. The *d-dim* is the key element for controlling the privacy level where the smaller the *d-dim* value, the higher privacy level of the local concealment algorithm. However, clearly the highest privacy is at *d-dim*=100. There is a noticeable drop of attained privacy when we change *d-dim* from 300 to 600. The *d-dim* value 400 is considered as a critical point for privacy. Regarding the global concealment algorithm, the relationship between different Hilbert curve parameters (order and step length) on the accuracy and privacy levels attained was measured. The locally concealed dataset was mapped to Hilbert values using order 3, 6, and 9. The step length was gradually increased from 10 to 80.

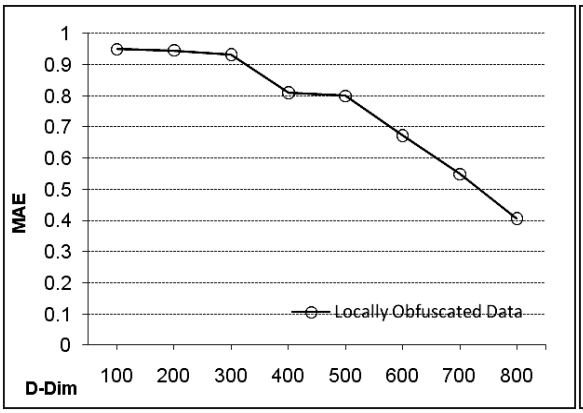


Figure 3.27: Accuracy for the concealed dataset using local concealment algorithm

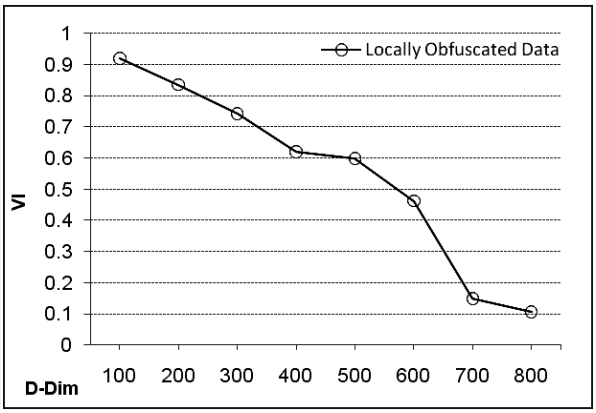


Figure 3.28: Privacy levels for the concealed dataset using local concealment algorithm

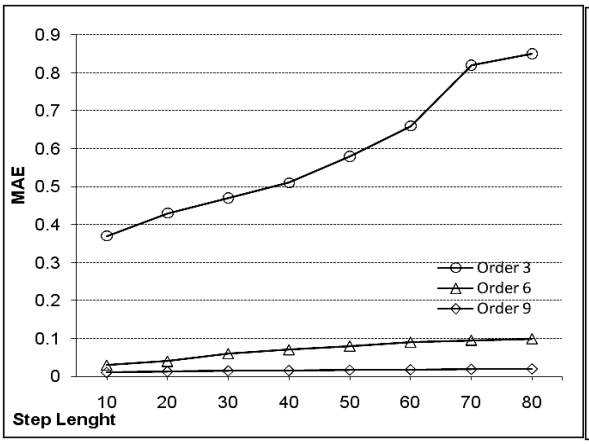


Figure 3.29: Accuracy level for different step length and orders for global concealment algorithm

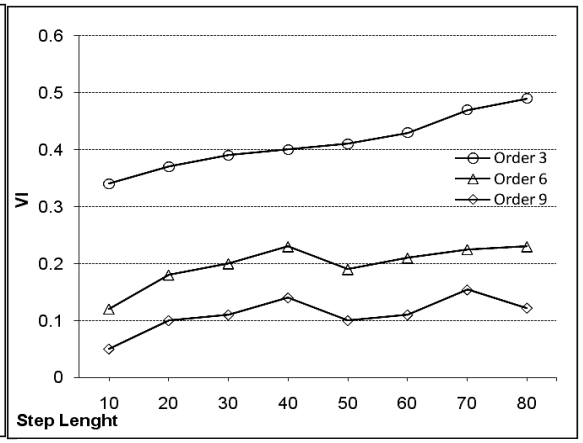


Figure 3.30: Privacy level for different step length and orders for global concealment algorithm

Figure 3.29 shows the accuracy of recommendations based on the different step length and curve order. As seen, when the order increases, the concealed data can offer better predictions for the ratings. This is because when the order has a higher value, the granularity of the Hilbert curve becomes finer. Therefore, the mapped values can preserve the data distribution of the original dataset. However, selecting a larger step length increases the accuracy values as large partitions are formed with a higher range to generate random values from it, such that these random values substitute real values in the dataset. Finally, as shown in Figure 3.30, when the order increases, a smaller range is calculated

within each partition which introduces fewer substituted values compared with lower orders that attain higher variation of information (VI) values. The reason for this is that the larger order divides the *m-dimensional* profile into more grids, which makes the Hilbert curve better at reflecting the data distribution. We can also see that for the same Hilbert curve order, the VI values are generally the same for the different step length except for order 3, in which VI values have a sharp increase when the step length grows from 50 to 60. The effect of increasing step length on VI values is more sensible in lower curve orders as fewer girds are formed and the increase of the step length covers more portions of them, which will introduce a higher range to generate the random values from it. The target user should select the parameters of the global concealment algorithm in such a way as to achieve a trade-off between privacy and accuracy. The various approaches which were taken for attaining users' profiles privacy along with the set of PETs to achieve this goal are described in more detail in Articles IX and X.

3.8 Privacy Aware Location based Recommender Service

In this section, an overview of the scenario was presented where a location aware recommender service (LARS) is implemented on an external third party server and end-users give their location information to that server in order to receive useful recommendations in various areas such as travel information, shopping, entertainment, taxi services and location based advertising. One major concern about the end-users' adoption of this new service lies in privacy concerns of the users, which in most cases, prevent them from fully embracing this service. Location disclosure due to insider attacks at the service provider side is another privacy concern for most of the end-users. *EMCP* allows privacy aware users to detect nearby points-of-interest without revealing their real position.

There are two possible ways for the user's disclosure: through his/her location information or through the user's network address (IP). *EMCP* employs two principles to eliminate

these two disclosure channels, respectively. The two stage concealment process was used to conceal the user's location information and an anonymous data collection protocol is used to hide the user's network identity by routing the communication with other participants through relaying nodes in Tor's anonymous network [138]. The recommendation process based on the two stage concealment process in our framework can be summarized as following:

1. The policy agent acquires the user's privacy preferences and expresses them using APPEL as a set of preferences rules which are then decoded into a set of elements that are stored in a "privacy preferences" database. These rules contain both a privacy policy and an action to be taken for such a privacy policy. This will enable the preference checker to make self-acting decisions on various elements that are encountered during the data collection process regarding different requests. This is an essential step in order to assure that the released data does not violate the privacy of its owner.
2. The target user broadcasts a message to other nearby users to start a recommendation process to obtain referrals regarding certain contexts. Moreover, he/she runs the local concealment process on his /her released request.
3. Individual users that decide to respond to that request from peer-groups then elect a super-peer within each peer-group. Each participant within the peer-group executes the local concealment process on his/her profile's preferences. The participants then submit their locally concealed profiles' preferences either to the requester or to the super-peer (trusted aggregator) of their peer-group.
4. The trusted aggregator aggregates the collected preferences then executes a global concealment process on the aggregated group profile.
5. The trusted aggregator submits the globally concealed group profile together with pseudonyms of users who participate in the recommendation process to the location aware recommender service (LARS).

6. The LARS inserts the received data in its database then updates its model using the received group profile and extracts the referrals list for this group profile. Finally, LARS submits this list to the trusted aggregator.
7. The trusted aggregator distributes this list to the other members of peer-groups in order to maintain a good reputation between them.

Evaluation Results

It is necessary to evaluate the impact of adding location data as a part of participants' profiles on the accuracy of the generated referrals. Similarity scores between each pair of participants are used by super-peers to compute sub-communities of related participants. When calculating scores, participants take into account two factors, the similarity value between their preferences/interests and distance between their locations. The final similarity scores are weighted and summed based on these two values.

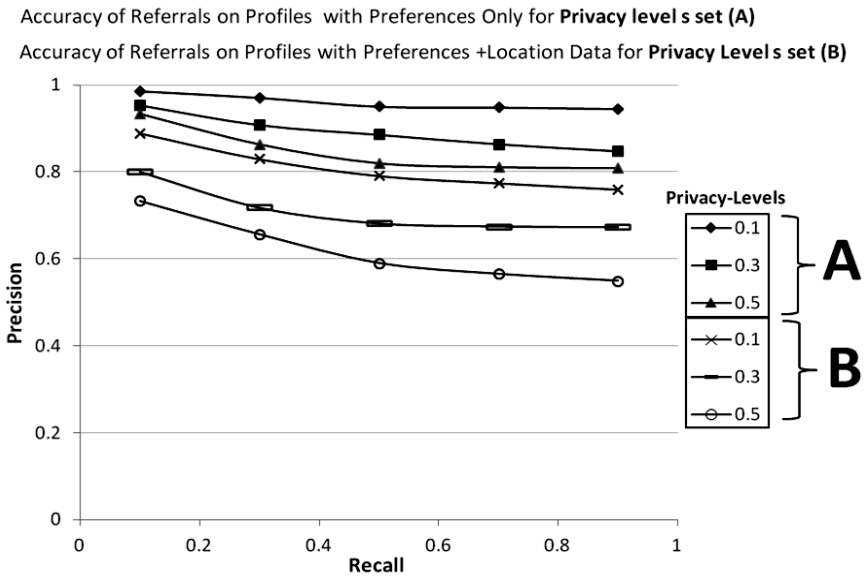


Figure 3.31: Accuracy of referrals when combine location+ preferences data with different privacy levels

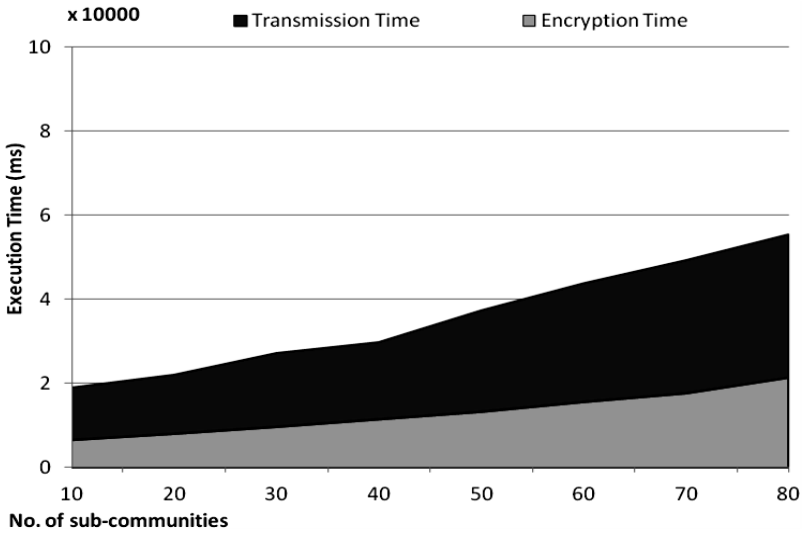


Figure 3.32: Execution Time for extracting different sub-communities

Figure 3.31 depicts the comparative results of running proposed protocols with varying privacy levels of the two stage concealment process on a dataset containing both preferences and location data. For the same dataset, we run this experiment on preferences data only. As shown in Figure 3.31, the accuracy of referrals generated with preferences data in profiles only achieve higher accuracy than ones with preferences and location data, since the search for suitable sub-communities for referrals will be performed across whole sub-communities representatives. On the other hand, using both preferences and location data in participants' profiles constrains the underlying search space for generating referrals which affect accuracy, since the recommendations process will only be performed on nearby sub-communities representatives only. However, this does not downgrade the accuracy of the provided referrals, since the local concealment algorithm utilizes a cloaking strategy around participant and sub-communities locations in order to preserve location privacy, which in turn increases the possibility for each participant to have an abundance of nearby sub-communities representatives. Adding location data enhances the quality of provided service with location awareness and guiding facilities and reduces the time required to generate referrals. In the second experiment performed on the two stage concealment process, the efficiency of our solution with increasing number of sub-

communities was measured. The execution time in terms of encryption and transmission time for participants' profiles was used. As seen from Figure 3.32, the proposed solution requires more communication as a consequence of the distributed design and communication needs for the two stage concealment process. This acceptable overhead is shared among all participants while the benefit is to protect their privacy without hampering referral quality. The processes of selecting super-peers and generating distributed keys are done once in the setup time before the start of the protocols, so the required time for them can be omitted.

3.9 Answers to the Research Question

This thesis is believed to make significant contributions towards immune privacy enhancing technologies. In this section, the answers to the research questions will be presented. This will be followed by Table 3.1 that maps the achievement to the research questions that they address and the papers that serve as the output of this research.

Q1. How can we build a clustering model on data distributed between multiple sites bearing in mind the privacy issue? Furthermore, how can we measure the validity of this clustering model in practical scenarios?

Answer: This question is addressed in Articles I, II, XIV, XV, and XVI. We design distributed clustering algorithms that attain clustering based on two main steps: (1) extracting local statistical patterns from the data at each site and then aggregating them, and (2) computing the global clustering model based on these statistical patterns. The proposed clustering algorithm produces accurate clusters with arbitrary shapes, sizes, and densities over the distributed dataset. Moreover, it can be utilized in privacy preserving scenarios as the model building retrieves the original statistical properties without the need to collect the entire original data from each site. DLC employs various objective functions within two consecutive steps in order to detect

global clusters across all sites that provide an optimal solution for these functions. The first step is used to detect local dense regions or clusters on each site independently. The final step is used to create global dense regions or clusters by merging all of the discovered local dense regions. Applications were presented to demonstrate the effectiveness of the proposed distributed clustering algorithm in solving sensible real world problems. DLC have been used in two scenarios, one related to personalized medical support for cardio-vascular monitoring and the other related to the complex search approach in data mash-up service.

Q2. What threat models can be utilized in third party social recommender services when users release their real profiles to earn accurate referrals? Furthermore, can OECD privacy principles be elicited for developing practical PETs for social recommender services?

Answer: This question is addressed in Articles III, IV, VII, VIII, VI, XI, XII and V. We assume that an adversary aims to collect preferences in user's profiles to identify and track users. Thus, we consider our main adversary to be an untrusted recommender service to which users send their preferences. We do not assume the social recommender service to be completely malicious. This is a realistic assumption because the social recommender service needs to accomplish some business goals and increase its revenues. The social recommender service can construct the profiles of the users based on the requests sent. Hence, the problem we are tackling has two sides. We want to detain the ability of the adversary to identify users based on a set of identifying interests and thus track them by correlating these data with data from other publicly accessible databases or leaked datasets and at the same time we want to prevent the adversary from profiling the users through their network identity and therefore invade their privacy. Intuitively, the system privacy is high if the social recommender service is not able to reconstruct the real users' preferences based on

the information available to it. We utilized the Graph Matching attack proposed in [126] and the Re-identification attack proposed in [127] to measure the difficulty of finding correct matches between the original dataset and its concealed version. A list of appropriate legal requirements that serve as "design criteria" according to which PETs for social recommender services can be proposed was presented in Section 3.2.1.

Q3. Without the need to trust the provider's declared policies and self-regulations, what framework can support protecting users' privacy before their data is shared with the social recommender services such that this framework attains privacy and anonymity for participants? Furthermore, what practical scenarios can benefit from the whole architecture?

Answer: This question is addressed in Articles VI, VII, VIII, IX, XI, and XXI. The data assemblage of the social recommender service providers divulges the sensitive data in the profiles of many users. This can give a rise to privacy breaches for these profiles and thus a need for privacy enhancing technologies to protect their privacy. This research aimed to give the users complete control over his/her personal profiles which are stored only on his/her own device. Focusing on this aim, we proposed a collaborative privacy framework that facilitated the privacy commitment by utilizing the social side of the recommender services in order to perform a two stage concealment process based on stochastic techniques to conceal the users' data and thus reduce privacy risks. The proposed framework was applied as a middleware equipped with all these stochastic techniques to make it possible to attain our aim. The proposed middleware enables participants to be organized in a distributed topology, where participants are organized into peer-groups and each peer-group contains a reliable peer to act as a trusted aggregator that is an entitled super-peer who will be responsible for anonymously sending the aggregated data of members

within this peer-group to the social recommender service. This topological formation prevents the service provider from creating a centralized database with raw personal data from each user and permits a decentralized execution of a two-stage concealment process on users' personal data. The proposed framework was utilized in diverse scenarios to create privacy aware versions for three beneficial applications of the social recommender service, which are a recommender service for IPTV content providers, data mash-up service for IPTV recommender services and community discovery and recommendation service. Privacy aware versions of location based recommendation service and mobile jukebox content recommender service were also introduced in order to show the applicability of our approach. The implementation and evaluation of such applications of the collaborative privacy framework confirmed that it is possible to employ the personal profiles of users while preserving their privacy.

Q.4 What framework can support privacy in collaborative platforms such that a recommender service can leverage the databases of different competing online database providers to provide better referrals without breaching the privacy of their users? Furthermore, what application can benefit from the whole architecture?

Answer: This question is addressed in Articles X, XIII, and XVI. We refine the scenario of the collaborative privacy framework in social recommender services to collaboration based data mash-up service. The data mash-up service integrates datasets of users' preferences from multiple online movie database providers for a recommender service running recommendations for different IPTV providers. We assumed that the item set stored at each movie database provider is the same but the users registered at these providers are not identical. Additionally, we assumed that the users' preferences are stored in plain form at these movie database providers with which the users registered. This resulted in executing a two stage concealment

process between the online movie database providers and the data mash-up service. The need for the formation of peer-groups with various super-peers is eliminated as the data mash-up service will be acting as the one super-peer with a peer-group of movie database providers. The data mash-up service will execute a global concealment process closely to protect its reputation while the movie database providers will execute the local concealment process to preserve the privacy of their users' personal data. From the point of view of the social recommender service, the whole two stage concealment process is pre-executed. The recommender service sends a query to the data mash-up service to retrieve datasets related to that query in order to be used in enhancing its recommendations accuracy.

Q5. How can privacy be enhanced in third party social recommender services by technical means with a reasonable trade-off between privacy protection and accuracy loss? Can these technical means transform the original data into a new one that conceals sensitive data while preserving the required patterns for an accurate recommendation task? *Furthermore, can we develop a non-cryptography based technical means for multi-party recommendation problems so that existing traditional cryptography based recommendation algorithms can be used*?

Answer: This question is addressed in Articles III, IV, V, VI, VII, VIII, IX, XI, XII, and XV. Depending on the machine learning clustering analysis, we designed a clustering technique as a pattern preserving mechanism, which was used as the building block for the proposed stochastic techniques of the two stage concealment process in order to preserve the utility of data. Moreover, utilizing the clustering analysis aids in mitigating certain attacks on the data, which is concealed using our proposed techniques. As a result, the accuracy of extracted referrals is maintained while preserving the privacy of the preferences' data provided by participants. The proposed PETs are executed in two consecutive steps within a two stage concealment

process. The proposed PETs destroy the structure in the data but, at the same time, maintain some properties in it which is required in the planned recommendation. The implementation of these applications confirmed that it is feasible to make use of and, at the same time, to protect the personal sensitive data of individuals, and to do so in an accurate way. We reduce the reliance on secure multiparty computation protocols as a privacy enhancing mechanism since they are costly in terms of communication and execution considering the limited hardware resources in users' devices. Employing the proposed stochastic techniques as a pre-processing step before encrypting the users' data for a secure multi-party recommendation, adds an extra layer of secrecy for these algorithms without utilizing larger key sizes. As a result, this allows the secure multi-party recommendation to handle a big dataset efficiently.

Research Question	Challenges	Achievements	Article	Appendix
Q1	C1 and C4	A1	Article I	Appendix A
			Article II	
			Article XIV	Appendix E
			Article XV	
		A1and A4	Article XVI	
Q2	C1 and C2	A2, A3, and A5	Article III	Appendix B
			Article IV	
			Article VII	
			Article V	
			Article VIII	
			Article VI	
		A2, A3, and A6	Article XI	Appendix D
		A3	Article X	Appendix C
Q3	C1, C4, and C6	A2, A3, and A5	Article VI	Appendix B
			Article VII	
			Article VIII	
			Article V	
		A3	Article X	Appendix C
			Article XIII	Appendix D
Q4	C1, C2, C3, C4, and C5	A2, A3, and A4	Article X	Appendix C
			Article XIII	Appendix D
		A1 and A4	Article XVI	Appendix E

Research Question	Challenges	Achievements	Article	Appendix
Q5	C2,C3, And C5	A2, A3, and A5	Article III	Appendix B
			Article IV	
			Article V	
			Article VI	
			Article VII	
			Article VIII	
		A3	Article IX	Appendix C
		A3 and A4	Article XI	Appendix D
			Article XII	
		A2, A3, and A6	Article XIV	Appendix E
			Article XV	
			Article XVI	

Table 3.1: Research Questions, Challenges, Achievements, and Publications

Chapter 4

Concluding Remarks & Future Work

In this chapter, Section 4.1 includes the concluding remarks of the research that has been presented in this thesis, and Section 4.2, concludes this thesis by presenting the possible directions to continue this research.

4.1 Summary and Concluding Remarks

This thesis has investigated why privacy is needed and how it could be enhanced in social recommender services. Although the emphasis has been on the social recommender services, privacy in distributed clustering has also been studied.

This work proposed the usage of a collaborative privacy framework in order to create beneficial social recommender services. The proposed applications utilize the personal sensitive data of users while ensuring their privacy. We have also illuminated the importance of taking into account the underlying coalition when designing and deploying PETs for providing the users with anonymity and data privacy. The proposed PETs are analysed in terms of privacy and accuracy, and they are encouraged with real data-based experiments using off-the-shelf recommendation techniques on the concealed data.

Overall, this work is based on a theoretical approach and confirmed with experimental results of the prototype implementations. Promising results were obtained, which clearly indicate that the proposed solution can enable online social recommender services to collect concealed data and generate accurate referrals without compromising the privacy of their users. However, privacy and accuracy are conflicting goals, so to obtain a balance between accuracy and privacy, the parameters of the proposed PETs have to be adjusted. According to the experiments' results, the proposed approach parameters have different effects on

privacy and accuracy. A functional solution developed based on collaborative privacy is appropriate for use in various social recommender services scenarios.

4.2 Future Work

Future research needs to study how selective concealment of profiles instead of policy based concealment can affect the recommendations' results. In addition, our presented work [139, 140] is based on the hypothesis that dependency between trust and privacy can be depicted as an exponential function, but other models of this dependency can also be adopted depending on, for example, use case scenario. Thus, we aim to study and suggest different privacy-trust correlation models based on the requirements in hand.

Electing super-peers is based on a global success-trust dependency but a possible new dimension could envision expressing this relation for each user independently without the need for a trusted third party. This would provide a more accurate representation of the trusted super-peer, not influenced as much by the dominant users in the system. Moreover, in all of the applications, users' trustworthiness is out of interest. Considering malicious user existence would generate interesting discussions.

A more thorough evaluation of our approach would be useful, such as case studies on a small or large scale. Furthermore, it would be appealing to investigate other innovative applications, which can be used in everyday life, with emphasis on the users' privacy, where personal data is kept on the user's side. This will give the users an opportunity to control and protect their personal data. Finally, this thesis presented the general directions for future research in the area of the users' data privacy which will continue to be a challenge in new information technologies.

References

1. Wen, J., Anthony, K.H.T., Jiawei, H., Wei, W.: Ranking outliers using symmetric neighborhood relationship. Springer (2006)
2. Jr: Hierarchical Grouping to Optimize an Objective Function. Journal of the American Statistical Association **58** (1963) 236-244
3. Ammar, Z.S., Gaber, M.M.: DDG-Clustering: A Novel Technique for Highly Accurate Results. IADIS European Conference on Data Mining (ECDM 2009), Algarve, Portugal (2009)
4. Olson, J.S., Grudin, J., Horvitz, E.: A study of preferences for sharing and privacy. CHI '05 extended abstracts on Human factors in computing systems. ACM, Portland, OR, USA (2005) 1985-1988
5. Solove, D.J.: The digital person: Technology and privacy in the information age, Vol. 1. NYU Press (2004)
6. Culnan, M.J., Williams, C.C.: How ethics can enhance organizational privacy: lessons from the choicepoint and TJX data breaches. Mis Quarterly **33** (2009) 673-687
7. Jones, A., Dardick, G.S., Davies, G., Sutherland, I.: 2008 Analysis of Information Remaining on Disks Offered For Sale on the Second Hand Market, The. J. Int'l Com. L. & Tech. **4** (2009)
8. Adar, E.: User 4xxxxx9: Anonymizing query logs. Proc of Query Log Analysis Workshop, International Conference on World Wide Web (2007)
9. Warren, S.D., Brandeis, L.D.: The right to privacy. Harvard law review **4** (1890) 193-220
10. Westin, A.F.: Privacy and freedom. Washington and Lee Law Review **25** (1968) 166
11. Margulis, S.T.: On the Status and Contribution of Westin's and Altman's Theories of Privacy. Journal of Social Issues **59** (2003) 411-429
12. Lederer, S., Hong, I., Dey, K., Landay, A.: Personal privacy through understanding and action: five pitfalls for designers. Personal Ubiquitous Comput. **8** (2004) 440-454
13. Ferrari, E., Press, I.: Web and information security. IRM Press (2007)
14. Thuraisingham, B.: Data mining, national security, privacy and civil liberties. SIGKDD Explor. Newsl. **4** (2002) 1-5
15. Chaum, D.: Security without identification: Transaction systems to make big brother obsolete. Communications of the ACM **28** (1985) 1030-1044
16. Gartner: Security concerns to stunt e-commerce growth. ComputerWorld (2005)
17. Cranor, L.F., Reagle, J., Ackerman, M.S.: Beyond Concern: Understanding Net Users' Attitudes About Online Privacy. CoRR **cs.CY/9904010** (1999)
18. Teltzrow, M., Kobsa, A.: Impacts of user privacy preferences on personalized systems: a comparative study. Designing personalized user experiences in eCommerce. Kluwer Academic Publishers (2004) 315-332
19. Directive, E.: 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data. Official Journal of the EC **23** (1995) 6

20. Commission, E.: Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector. Official Journal L **201** (2002) 07
21. Cockcroft, S.K.S., Clutterbuck, P.J.: Attitudes towards information privacy. School of Multimedia and Information Technology, Southern Cross University (2001)
22. Halevy, A., Norvig, P., Pereira, F.: The unreasonable effectiveness of data. Intelligent Systems, IEEE **24** (2009) 8-12
23. Kaplan, R.S., Norton, D.P.: Using the balanced scorecard as a strategic management system. Harvard business review **74** (1996) 75-85
24. Zamir, O., Etzioni, O.: Web document clustering: A feasibility demonstration. Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval. ACM (1998) 46-54
25. Kumar, M., Patel, N.R., Woo, J.: Clustering seasonality patterns in the presence of errors. Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM (2002) 557-563
26. Schafer, J.B., Konstan, J.A., Riedl, J.: E-commerce recommendation applications. Applications of Data Mining to Electronic Commerce. Springer (2001) 115-153
27. DuBois, T., Golbeck, J., Kleint, J., Srinivasan, A.: Improving recommendation accuracy by clustering social networks with trust. Recommender Systems & the Social Web (2009) 1-8
28. Berkhin, P.: A survey of clustering data mining techniques. Grouping multidimensional data. Springer (2006) 25-71
29. Forman, G., Zhang, B.: Distributed data clustering can be efficient and exact. ACM SIGKDD explorations newsletter **2** (2000) 34-38
30. Huang, Z., Du, W., Chen, B.: Deriving private information from randomized data. Proceedings of the 2005 ACM SIGMOD international conference on Management of data. ACM, Baltimore, Maryland (2005) 37-48
31. Kargupta, H., Datta, S., Wang, Q., Sivakumar, K.: On the Privacy Preserving Properties of Random Data Perturbation Techniques. Proceedings of the Third IEEE International Conference on Data Mining. IEEE Computer Society (2003) 99
32. Ramakrishnan, N., Keller, B.J., Mirza, B.J., Grama, A.Y., Karypis, G.: Privacy risks in recommender systems. IEEE Internet Computing **5** (2001) 54-63
33. Daswani, N., Garcia-Molina, H., Yang, B.: Open problems in data-sharing peer-to-peer systems. Database Theory—ICDT 2003. Springer (2002) 1-15
34. Cranor, L.F.: 'I didn't buy it for myself' privacy and ecommerce personalization. Proceedings of the 2003 ACM workshop on Privacy in the electronic society. ACM, Washington, DC (2003)
35. Leeuwen, J.: Handbook of Theoretical Computer Science: Algorithms and complexity. Volume A, Vol. 1. Access Online via Elsevier (1990)
36. Damgard, I., Geisler, M., Kroigard, M.: Homomorphic encryption and secure comparison. International Journal of Applied Cryptography **1** (2008) 22-31

37. Gentry, C.: Computing arbitrary functions of encrypted data. Communications of the ACM **53** (2010) 97-105
38. Fan, J., Vercauteren, F.: Somewhat Practical Fully Homomorphic Encryption. IACR Cryptology ePrint Archive **2012** (2012) 144
39. Levin, L.A.: The tale of one-way functions. Problems of Information Transmission **39** (2003) 92-103
40. Gionis, A., Indyk, P., Motwani, R.: Similarity search in high dimensions via hashing. VLDB, Vol. 99 (1999) 518-529
41. Rivest, R.L., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM **21** (1978) 120-126
42. ElGamal, T.: A public key cryptosystem and a signature scheme based on discrete logarithms. Information Theory, IEEE Transactions on **31** (1985) 469-472
43. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. Advances in cryptology—EUROCRYPT'99. Springer (1999) 223-238
44. Fuchsbauer, G.J.: An Introduction to Probabilistic Encryption. Osječki matematički list **6** (2006) 37-44
45. Cramer, R., Shoup, V.: A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. Advances in Cryptology—CRYPTO'98. Springer (1998) 13-25
46. Solo, D., Housley, R., Ford, W.: Internet X. 509 public key infrastructure certificate and CRL profile. (1999)
47. Rescorla, E.: SSL and TLS: designing and building secure systems, Vol. 1. Addison-Wesley Reading (2001)
48. Yao, A.C.: Protocols for secure computations. Proceedings of the 23rd Annual Symposium on Foundations of Computer Science. IEEE Computer Society (1982) 160-164
49. ECRYPTII: Final Report on Main Computational Assumptions in Cryptography. ECRYPT (2013)
50. Elmisery, A., Huaiguo, F.: Privacy Preserving Distributed Learning Clustering Of HealthCare Data Using Cryptography Protocols. 34th IEEE Annual International Computer Software and Applications Workshops (COMPSACW), Seoul, South Korea (2010) 140-145
51. Han, J., Kamber, M., Pei, J.: Data mining: concepts and techniques. Morgan kaufmann (2006)
52. Zamir, O., Etzioni, O.: Grouper: a dynamic clustering interface to Web search results. Computer Networks **31** (1999) 1361-1374
53. Kaufman, L., Rousseeuw, P.J.: Finding groups in data: an introduction to cluster analysis, Vol. 344. John Wiley & Sons (2009)
54. Ng, R.T., Han, J.: CLARANS: A method for clustering objects for spatial data mining. Knowledge and Data Engineering, IEEE Transactions on **14** (2002) 1003-1016
55. Jain, A.K., Dubes, R.C.: Algorithms for clustering data. Prentice-Hall, Inc. (1988)
56. Xu, R., Wunsch, D.: Survey of clustering algorithms. Neural Networks, IEEE Transactions on **16** (2005) 645-678

57. Ester, M., Kriegel, H.-p., Jörg, S., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. Proc. of 2nd International Conference on Knowledge Discovery and Data Mining (1996) 226-231
58. Ankerst, M., Breunig, M.M., Kriegel, H.-P., J\, \#246, Sander, r.: OPTICS: ordering points to identify the clustering structure. Proceedings of the 1999 ACM SIGMOD international conference on Management of data. ACM, Philadelphia, Pennsylvania, United States (1999).
59. Wang, W., Yang, J., Muntz, R.R.: STING: A Statistical Information Grid Approach to Spatial Data Mining. Proceedings of the 23rd International Conference on Very Large Data Bases. Morgan Kaufmann Publishers Inc. (1997) 186-195
60. Sheikholeslami, G., Chatterjee, S., Zhang, A.: Wavecluster: A multi-resolution clustering approach for very large spatial databases. VLDB, Vol. 98 (1998) 428-439
61. Cheeseman, P., Stutz, J.: Bayesian classification (AutoClass): theory and results. Advances in knowledge discovery and data mining. American Association for Artificial Intelligence (1996) 153-180
62. Eisenhardt, M., Müller, W., Henrich, A.: Classifying Documents by Distributed P2P Clustering. GI Jahrestagung (2) **35** (2003) 286-291
63. Bandyopadhyay, S., Giannella, C., Maulik, U., Kargupta, H., Liu, K., Datta, S.: Clustering distributed data streams in peer-to-peer environments. Information Sciences **176** (2006) 1952-1985
64. Datta, S., Bhaduri, K., Giannella, C., Wolff, R., Kargupta, H.: Distributed data mining in peer-to-peer networks. Internet Computing, IEEE **10** (2006) 18-26
65. Strehl, A., Ghosh, J.: A scalable approach to balanced, high-dimensional clustering of market-baskets. High Performance Computing—HiPC 2000. Springer (2000) 525-536
66. Ricci, F., Rokach, L., Shapira, B.: Introduction to recommender systems handbook. Recommender Systems Handbook. Springer (2011) 1-35
67. Jannach, D., Zanker, M., Felfernig, A., Friedrich, G.: Recommender systems: an introduction. Cambridge University Press (2010)
68. Konstan, J.A.: Introduction to recommender systems: Algorithms and evaluation. ACM Transactions on Information Systems (TOIS) **22** (2004) 1-4
69. Jeckmans, A.J., Beye, M., Erkin, Z., Hartel, P., Lagendijk, R.L., Tang, Q.: Privacy in Recommender Systems. Social Media Retrieval. Springer (2013) 263-281
70. Van Meteren, R., Van Someren, M.: Using content-based filtering for recommendation. Proceedings of the Machine Learning in the New Information Age: MLnet/ECML2000 Workshop (2000)
71. Zhang, T., Iyengar, V.S.: Recommender systems using linear classifiers. The Journal of Machine Learning Research **2** (2002) 313-334
72. Mladenic, D.: Personal WebWatcher: design and implementation. (1996)
73. King, A., Lakhani, K.R.: Using Open Innovation to Identify the Best Ideas.
74. Hansell, S.: Why Yelp works. New York Times **12** (2008)

75. Chen, Y., Xie, J.: Online consumer review: Word-of-mouth as a new element of marketing communication mix. Management Science **54** (2008) 477-491
76. Pazzani, M.J.: A framework for collaborative, content-based and demographic filtering. Artificial Intelligence Review **13** (1999) 393-408
77. Vozalis, M., Margaritis, K.G.: On the enhancement of collaborative filtering by demographic data. Web Intelligence and Agent Systems **4** (2006) 117-138
78. Wahlster, W., Kobsa, A.: User models in dialog systems. Springer (1989)
79. Krulwich, B.: Lifestyle finder: Intelligent user profiling using large-scale demographic data. AI magazine **18** (1997) 37
80. Burke, R.: Hybrid web recommender systems. The adaptive web. Springer (2007) 377-408
81. Giesler, M., Pohlmann, M.: The social form of Napster: Cultivating the paradox of consumer emancipation. Advances in consumer research **30** (2003) 94-100
82. Gunes, I., Kaleli, C., Bilge, A., Polat, H.: Shilling attacks against recommender systems: a comprehensive survey. Artificial Intelligence Review (2012) 1-33
83. Garber, L.: Denial-of-service attacks rip the Internet. IEEE Computer **33** (2000) 12-17
84. Canny, J.: Collaborative Filtering with Privacy. Proceedings of the 2002 IEEE Symposium on Security and Privacy. IEEE Computer Society (2002) 45
85. Canny, J.: Collaborative filtering with privacy via factor analysis. Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, Tampere, Finland (2002) 238-245
86. Esma, A.: Experimental Demonstration of a Hybrid Privacy-Preserving Recommender System. In: Gilles, B., Jose, M.F., Flavien Serge Mani, O., Zbigniew, R. (eds.), Vol. 0 (2008)
87. Polat, H., Du, W.: Privacy-Preserving Collaborative Filtering Using Randomized Perturbation Techniques. Proceedings of the Third IEEE International Conference on Data Mining. IEEE Computer Society (2003) 625
88. Polat, H., Du, W.: SVD-based collaborative filtering with privacy. Proceedings of the 2005 ACM symposium on Applied computing. ACM, Santa Fe, New Mexico (2005) 791-795
89. Miller, B.N., Konstan, J.A., Riedl, J.: PocketLens: Toward a personal recommender system. ACM Trans. Inf. Syst. **22** (2004) 437-476
90. Fung, B., Trojer, T., Hung, P.C., Xiong, L., Al-Hussaeni, K., Dssouli, R.: Service-oriented architecture for high-dimensional private data mashup. Services Computing, IEEE Transactions on **5** (2012) 373-386
91. Chris, C., Don, M.: Security and Privacy Implications of Data Mining. (1996)
92. Du, W., Atallah, M.J.: Secure multi-party computation problems and their applications: a review and open problems. Proceedings of the 2001 workshop on New security paradigms. ACM, Cloudcroft, New Mexico (2001) 13-22
93. Andrew Chi-Chih, Y.: How to generate and exchange secrets. Vol. 0 (1986) 162-167
94. Yehuda, L., Benny, P.: Secure Multiparty Computation for Privacy-Preserving Data Mining. (2008)

95. Vaidya, J., Clifton, C.: Privacy preserving association rule mining in vertically partitioned data. Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, Edmonton, Alberta, Canada (2002) 639-644
96. Murat, K.: Privacy-Preserving Distributed Mining of Association Rules on Horizontally Partitioned Data. IEEE Transactions on Knowledge and Data Engineering **16** (2004) 1026-1037
97. Vaidya, J., Clifton, C.: Privacy-preserving k-means clustering over vertically partitioned data. Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, Washington, D.C. (2003) 206-215
98. Brumen, B., Welzer, T., Družovec, M., Golob, I., Jaakkola, H., Rozman, I., Kubalík, J.: Protecting Medical Data for Decision-Making Analyses. Journal of Medical Systems **29** (2005) 65-80
99. Wang, J., Fukasawa, T., Urabe, S., Takata, T., Miyazaki, M.: Mining Frequent Patterns Securely in Distributed System. Oxford University Press (2006)
100. Merugu, S., Ghosh, J.: Privacy-preserving Distributed Clustering using Generative Models. Proceedings of the Third IEEE International Conference on Data Mining. IEEE Computer Society (2003) 211
101. Tendick, P., Matloff, N.: A modified random perturbation method for database security. ACM Trans. Database Syst. **19** (1994) 47-63
102. Dalenius, T., Reiss, S.P.: Data-swapping: A technique for disclosure control. Journal of Statistical Planning and Inference **6** (1982) 73-85
103. Agrawal, D., Aggarwal, C.C.: On the design and quantification of privacy preserving data mining algorithms. Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems. ACM, Santa Barbara, California, United States (2001) 247-255
104. Evfimievski, A.: Randomization in privacy preserving data mining. SIGKDD Explor. Newsl. **4** (2002) 43-48
105. Guo, L., Guo, S., Wu, X.: Privacy Preserving Market Basket Data Analysis. Proceedings of the 11th European conference on Principles and Practice of Knowledge Discovery in Databases. Springer-Verlag, Warsaw, Poland (2007) 103-114
106. Chen, K., Liu, L.: Privacy-Preserving Multiparty Collaborative Mining with Geometric Data Perturbation. IEEE Trans. Parallel Distrib. Syst. **20** (2009) 1764-1776
107. Chen, K., Sun, G., Liu, L.: Towards Attack-Resilient Geometric Data Perturbation. Vol. 127. Society for Industrial Mathematics (2007) 78
108. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertain. Fuzziness Knowl.-Based Syst. **10** (2002) 557-570
109. Samarati, P.: Protecting Respondents' Identities in Microdata Release. IEEE Trans. on Knowl. and Data Eng. **13** (2001) 1010-1027
110. Aggarwal, G., Feder, T., Kenthapadi, K., Motwani, R., Panigrahy, R., Thomas, D., Zhu, A.: Approximation Algorithms for k-Anonymity. Proceedings of the International Conference on Database Theory (ICDT 2005), Edinburgh, UK (2005)

111. LeFevre, K., DeWitt, D.J., Ramakrishnan, R.: Incognito: efficient full-domain K-anonymity. Proceedings of the 2005 ACM SIGMOD international conference on Management of data. ACM, Baltimore, Maryland (2005) 49-60

112. Machanavajjhala, A., Kifer, D., Gehrke, J., Venkitasubramaniam, M.: L-diversity: Privacy beyond k-anonymity. ACM Trans. Knowl. Discov. Data **1** (2007) 3

113. Ninghui, L., Tiancheng, L., Venkatasubramanian, S.: t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on (2007) 106-115

114. Johnsten, T., Raghavan, V.V.: A methodology for hiding knowledge in databases. Proceedings of the IEEE international conference on Privacy, security and data mining - Volume 14. Australian Computer Society, Inc., Maebashi City, Japan (2002) 9-17

115. Xie, Y., Johnsten, T., Raghavan, V.V.: Knowledge Hiding in Databases for concept-based data mining algorithms. Proceedings of the winter international synposium on Information and communication technologies. Trinity College Dublin, Cancun, Mexico (2004) 1-8

116. Dasseni, E., Verykios, V.S., Elmagarmid, A.K., Bertino, E.: Hiding Association Rules by Using Confidence and Support. Proceedings of the 4th International Workshop on Information Hiding. Springer-Verlag (2001) 369-383

117. Verykios, V.S., Elmagarmid, A.K., Bertino, E., Saygin, Y., Dasseni, E.: Association rule hiding. Knowledge and Data Engineering, IEEE Transactions on **16** (2004) 434-447

118. Wang, E.T., Lee, G., Lin, Y.T.: A Novel Method for Protecting Sensitive Knowledge in Association Rules Mining. Proceedings of the 29th Annual International Computer Software and Applications Conference - Volume 01. IEEE Computer Society (2005) 511-516

119. Xiao, X., Tao, Y.: Personalized privacy preservation. Proceedings of the 2006 ACM SIGMOD international conference on Management of data. ACM, Chicago, IL, USA (2006) 229-240

120. Oliveira, S.R., Zaïane, O.R.: Toward standardization in privacy-preserving data mining. In: 2004 (ed.): ACM SIGKDD 3rd Workshop on Data Mining Standards, Vol. 7

121. Goecks, J., Edwards, W.K., Mynatt, E.D.: Challenges in supporting end-user privacy and security management with social navigation. Proceedings of the 5th Symposium on Usable Privacy and Security. ACM, Mountain View, California (2009) 1-12

122. Hinneburg, A., Hinneburg, E., Keim, D.A.: An efficient approach to clustering in large multimedia databases with noise. Int. Conf. on Knowledge Discovery in Databases (1998)

123. Johnson, S.: Hierarchical clustering schemes. Psychometrika **32** (1967) 241-254

124. Kalton, A., Langley, P., Wagstaff, K., Yoo, J.: Generalized clustering, supervised learning, and data assignment. Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, San Francisco, California (2001) 299-304

125. Cuesta-Frau, D., Pérez-Cortés, J.C., Andreu-García, G.: Clustering of electrocardiograph signals in computer-aided Holter analysis. Computer Methods and Programs in Biomedicine **72** (2003) 179-196

126. Shokri, R., Pedarsani, P., Theodorakopoulos, G., Hubaux, J.-P.: Preserving privacy in collaborative filtering through distributed aggregation of offline profiles. Proceedings of the third ACM conference on Recommender systems. ACM (2009) 157-164

127. Domingo-Ferrer, J.: Record Linkage. In: Liu, L., ÖZsu, M.T. (eds.): Encyclopedia of Database Systems. Springer US (2009) 2353-2354

128. Kim, H.D.: Applying Consistency-Based Trust Definition to Collaborative Filtering. KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS **3** (2009) 366-374

129. Paillier, P.: Public-Key Cryptosystems Based on Composite Degree Residuosity Classes.

130. Damgård, I., Jurik, M.: A Generalisation, a Simpli.cation and Some Applications of Paillier's Probabilistic Public-Key System Public Key Cryptography. In: Kim, K. (ed.), Vol. 1992. Springer Berlin / Heidelberg (2001) 119-136

131. Damgård, I., Koprowski, M.: Practical Threshold RSA Signatures without a Trusted Dealer Advances in Cryptology — EUROCRYPT 2001. In: Pfitzmann, B. (ed.), Vol. 2045. Springer Berlin / Heidelberg (2001) 152-165

132. Boneh, D., Franklin, M.: Efficient generation of shared RSA keys

Advances in Cryptology — CRYPTO '97. In: Kaliski, B. (ed.), Vol. 1294. Springer Berlin / Heidelberg (1997) 425-439

133. Elmisery, A., Botvich, D.: Privacy Aware Obfuscation Middleware for Mobile Jukebox Recommender Services. The 11th IFIP Conference on e-Business, e-Service, e-Society. IFIP, Kaunas, Lithuania (2011)

134. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. **22** (2004) 5-53

135. Golle, P., McSherry, F., Mironov, I.: Data collection with self-enforcing privacy. Proceedings of the 13th ACM conference on Computer and communications security. ACM, Alexandria, Virginia, USA (2006) 69-78

136. Cuesta-Frau, D., Pérez-Cortés, J.C., Andreu-García, G.: Clustering of electrocardiograph signals in computer-aided Holter analysis. Computer methods and programs in biomedicine **72** (2003) 179-196

137. Parameswaran, R., Blough, D.M.: Privacy preserving data obfuscation for inherently clustered data. Int. J. Inf. Comput. Secur. **2** (2008) 4-26

138. Dingledine, R., Mathewson, N., Syverson, P.: Tor: the second-generation onion router. Proceedings of the 13th conference on USENIX Security Symposium - Volume 13. USENIX Association, San Diego, CA (2004) 303-320

139. Elmisery, A., Botvich, D.: Privacy Aware Recommender Service using Multi-agent Middleware- an IPTV Network Scenario. Informatica **36** (2012) 21-36

140. Elmisery, A., Botvich, D.: Enhanced Middleware for Collaborative Privacy in IPTV Recommender Services Journal of Convergence **2** (2011) 33-42

Part II

Included Papers

Appendix A: Distributed Clustering

Article I

A New Feature Weighted Fuzzy C-means Clustering Algorithm

Huaiguo Fu, Ahmed M. Elmisery

In Proceedings of the IADIS European Conference on Data Mining (ECDM 2009), Algarve, Portugal, June 2009.

A NEW FEATURE WEIGHTED FUZZY C-MEANS CLUSTERING ALGORITHM

Huaiguo Fu, Ahmed M. Elmisery
Telecommunications Software & Systems Group
Waterford Institute of Technology, Waterford, Ireland

ABSTRACT

In the field of cluster analysis, most of existing algorithms assume that each feature of the samples plays a uniform contribution for cluster analysis. Feature-weight assignment is a special case of feature selection where different features are ranked according to their importance. The feature is assigned a value in the interval [0, 1] indicating the importance of that feature, we call this value "feature-weight". In this paper we propose a new feature weighted fuzzy c-means clustering algorithm in a way which this algorithm be able to obtain the importance of each feature, and then use it in appropriate assignment of feature-weight. These weights incorporated into the distance measure to shape clusters based on variability, correlation and weighted features.

KEYWORDS

Cluster Analysis, Fuzzy Clustering, Feature Weighted.

1. INTRODUCTION

The Goal of cluster analysis is to assign data points with similar properties to the same groups and dissimilar data points to different groups [3]. Generally, there are two main clustering approaches i.e. crisp clustering and fuzzy clustering. In the crisp clustering method the boundary between clusters is clearly defined. However, in many real cases, the boundaries between clusters cannot be clearly defined. Some objects may belong to more than one cluster. In such cases, the fuzzy clustering method provides a better and more useful method to cluster these objects [2]. Cluster analysis has been widely used in a variety of areas such as data mining and pattern recognition [e.g.1, 4, 6]. Fuzzy c-means (*FCM*) proposed by [5] and extended by [4] is one of the most well-known methodologies in clustering analysis. Basically *FCM* clustering is dependent on the measure of distance between samples. In most situations, *FCM* uses the common Euclidean distance which supposes that each feature has equal importance in *FCM*. This assumption seriously affects the performance of *FCM*, so that the obtained clusters are not logically satisfying. Since in most real world problems, features are not considered to be equally important. Considering example in [17], the Iris database [9] which has four features, i.e., sepal length (*SL*), sepal width (*SW*), petal length (*PL*) and petal width (*PW*). Fig. 1 shows a clustering for Iris database based on features *SL* and *SW*, while Fig. 2 shows a clustering based on *PL* and *PW*. From Fig. 1, one can see that there are much more crossover between the star class and the point class. It is difficult for us to discriminate the star class from the point class. On the other hand, it is easy to see that Fig. 2 is more crisp than Fig. 1. It illustrates that, for the classification of Iris database, features *PL* and *PW* are more important than *SL* and *SW*. Here we can think of that the weight assignment (0, 0, 1, 1) is better than (1, 1, 0, 0) for Iris database classification.

ISBN: 978-972-8924-88-1 © 2009 IADIS

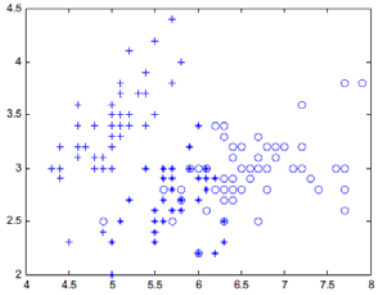


Figure1. Clustering Result of Iris Database Based on Feature Weights (1, 1, 0, 0) by *FCM* Algorithm

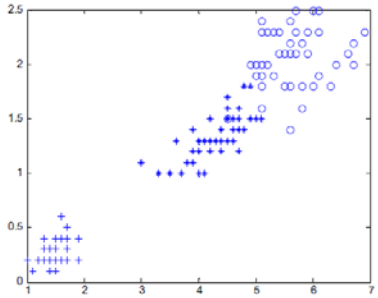


Figure 2. Clustering Result of Iris Database Based on Feature Weights (0, 0, 1, 1) by *FCM* Algorithm

Feature selection and weighting have been hot research topics in cluster analysis. Desarbo [8] introduced the *SYNCLUS* algorithm for variable weighting in *k*-means clustering. It is divided into two stages. First it uses *k*-means clustering with initial set of weights to partition data into *k* clusters. It then determines a new set of optimal weights by optimizing a weighted mean-square. The two stages iterate until they obtain an optimal set of weights.

Huang [7] presented *W-k*-means, a new *k*-means type algorithm that can calculate variable weights automatically. Based on the current partition in the iterative *k*-means clustering process, the algorithm calculates a new weight for each variable based on the variance of the within cluster distances. The new weights are used in deciding the cluster memberships of objects in the next iteration. The optimal weights are found when the algorithm converges. The weights can be used to identify important variables for clustering. The variables which may contribute noise to the clustering process can be removed from the data in the future analysis.

With respect to *FCM* clustering, it is sensitive to the selection of distance metric. Zhao [12] stated that the Euclidean distance give good results when all clusters are spheroids with same size or when all clusters are well separated. In [13, 10], they proposed a *G–K* algorithm which uses the well-known Mahalanobis distance as the metric in *FCM*. They reported that the *G–K* algorithm is better than Euclidean distance based algorithms when the shape of data is considered. In [11], the authors proposed a new robust metric, which is distinguished from the Euclidean distance, to improve the robustness of *FCM*.

Since *FCM*'s performance depends on selected metrics, it will depend on the feature-weights that must be incorporated into the Euclidean distance. Each feature should have an importance degree which is called feature-weight. Feature-weight assignment is an extension of feature selection [17]. The latter has only either 0-weight or 1-weight value, while the former can have weight values in the interval [0.1]. Generally speaking, feature selection method cannot be used as feature-weight learning technique, but the inverse is right. To be able to deal with such cases, we propose a new *FCM* Algorithm that takes into account weight of each feature in the data set that will be clustered. After a brief review of the *FCM* in section 2, a number of features ranking methods are described in section 3. These methods will be used in determining *FWA (feature weight assignment)* of each feature. In section 4 distance measures are studied and a new one is proposed to handle the different feature-weights. In section 5 we proposed the new *FCM* for clustering data objects with different feature-weights.

2. FUZZY C-MEAN ALGORITHM

Fuzzy *c*-mean (*FCM*) is an unsupervised clustering algorithm that has been applied to wide range of problems involving feature analysis, clustering and classifier design. *FCM* has a wide domain of applications such as agricultural engineering, astronomy, chemistry, geology, image analysis, medical diagnosis, shape analysis, and target recognition [14]. Unlabeled data are classified by minimizing an objective function based on a distance measure and clusters prototype. Although the description of the original algorithm dates back to 1974 [4, 5] derivatives have been described with modified definitions for the distance measure and prototypes for the cluster centers [12, 13, 11, 10] as explained above. The *FCM* minimizes an objective function J_m, which is the weighted sum of squared errors within groups and is defined as follows:

$$J_m(U, V; X) = \sum_{k=1}^{n} \sum_{i=1}^{n} u_{ik}^{m} \left\| x_k - v_i \right\|_A^2, 1 < m < \infty \quad \textbf{(1)}$$

Where V= $(v_1, v_2, \ldots\ldots, v_c)$ is a vector of unknown cluster prototype (centers) $v_i \in \Re^p$. The value of u_{ik} represent the grade of membership of data point x_k of set X= $\{x_1, x_2, \ldots\ldots, x_c\}$ to the *i*th cluster. The inner product defined by a distance measure matrix *A* defines a measure of similarity between a data object and the cluster prototypes. A hard fuzzy c-means partition of *X* is conveniently represented by a matrix $U = [u_{ik}]$. It has been shown by [4] that if $\left\| x_k - v_i \right\|_A^2 > 0$ for all *i* and *k*, then (U, V) may minimize J_m only, when *m>1* and

$$v_i = \frac{\sum_{k=1}^{n} (u_{ik})^m X_k}{\sum_{k=1}^{n} (u_{ik})^m} \quad \text{For } 1 \le i \le c \quad \textbf{(2)}$$

$$u_{ik} = \frac{1}{\sum_{j=1}^{c} \left(\frac{\left\| x_k - v_i \right\|_A^2}{\left\| x_k - v_j \right\|_A^2} \right)^{\frac{1}{m-1}}} \quad \text{For } 1 \le i \le c, 1 \le k \le n \quad \textbf{(3)}$$

Among others, J_m can be minimized by Picard iteration approach. This method minimizes J_m by initializing the matrix ***U*** randomly and computing the cluster prototypes *(Eq.2)* and the membership values *(Eq.3)* after each iteration. The iteration is terminated when it reaches a stable condition. This can be defined for example, when the changes in the cluster centers or the membership values at two successive iteration steps is smaller than a predefined threshold value.

The *FCM* algorithm always converges to a local minimum. A different initial guess of u_{ij} may lead to a different local minimum. Finally, to assign each data point to a specific cluster, defuzzification is necessary, e.g., by attaching a data point to a cluster for which the value of the membership is maximal [14].

3. ESTIMATING *FWA* OF FEATURES

In section 1 we mentioned that we propose a new clustering algorithm for a data objects with different feature-weights, which means that data with features of different *FWA* should be clustered. A key question that arises here is how we can determine the importance of each feature. In other words, we are about to assign a weight to each feature so that the weight of each feature determines the *FWA* of it.

To determine the *FWA* of features of a data set two major approaches can be adopted: **Human-based approach** and **Automatic approach**. In human-based approach we determine the *FWA* of each feature based on negotiation with an expert individual who has enough experience and knowledge in the field that is the subject of clustering. On the other hand, in automatic approach we use the data set itself to determine the *FWA* of its features. We will discuss more about these approaches in next lines.

Human-based approach: As is described above, in human-based approach by negotiating with an expert, we choose *FWA* of each feature. This approach has some advantages and some drawbacks. In some cases, using the data set itself to determine the *FWA* of each feature may fail to achieve the real *FWA*'s, and human-based approach should be adopted to determine the *FWA* of each feature. Fig.3 demonstrates a situation this case happens.

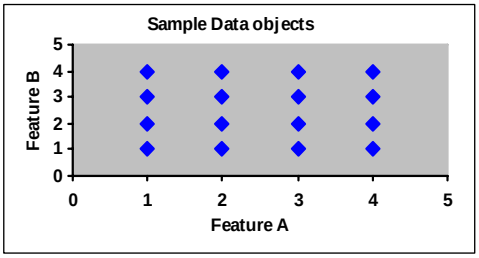


Figure 3. Data Object with Two Features

ISBN: 978-972-8924-88-1 © 2009 IADIS

Suppose Fig.3 shows a data objects in which *FWA* of feature ***A*** is two times *FWA* of feature ***B*** in reality. Since automatic approach uses the position of data points in the data space to determine the *FWA* of features, using data set itself to determine the *FWA* of features ***A*** and ***B*** (automatic approach) will lead to equal *FWA*'s for ***A*** and ***B***. Although this case (data set with homogeneously and equidistantly distributed data points) rarely happens in real world and is somehow an exaggerated one, it shows that, sometimes, human-base approach is the better choice.

On the other hand, human-based approach has its own drawbacks. We cannot guarantee that the behaviors that are observed by a human expert and used to determine the *FWA's* include all situations that can occur due to disturbances, noise, or plant parameter variations. Also suppose situation in which there is no human expert for negotiation to determine *FWA's.* How does this problem should be dealt with?

Structure the signal can be found using linear transforms. This approach does not take into account that the system has some structure. In the time domain, filtering is a linear transformation. The Fourier, Wavelet, and Karhunen-Loeve transforms have compression Capability and can be used to identify some structure in the signals. When we are using these transforms, we do not take into account any structure in the system.

Automatic approach: Several methods based on fuzzy set theory, artificial neural network, fuzzy-rough set theory, principle component analysis and neuro-fuzzy methods and have been reported [16] for weighted feature estimation. Some of the mentioned methods just rank features, but with some modifications they will be able to calculate the *FWA* of the features. Here we introduce a feature weight estimation method which can be used to determine the *FWA* of features. This method extends the one proposed in [15].

Let the *p*th pattern vector (each pattern is a single data item in the data set and a pattern vector is a vector which its elements are the values that the pattern features assume in the data set) be represented as

$$x^p = [x_1^p, x_2^p, \ldots\ldots\ldots, x_n^p] \quad \textbf{(4)}$$

Where *n* is the number of features of the data set, and x_i^p is the *i*th element of the vector. Let $prob_k$ and $d_k(x^p)$ stand for the priori probability for the class C_k and the distance of the pattern x^p from the *k*th mean vector,

$$m_k = [m_{k1}, m_{k2}, \ldots\ldots\ldots, m_{kn}] \quad \textbf{(5)}$$ respectively.

The feature estimation index for a subset (Ω) containing few of these *n* features is defined as

$$E = \sum_{x^p \in C_k} \sum_k \frac{s_k(x^p)}{\sum_{k' \neq k} s_{k'k}(x^p)} \times \alpha_k \quad \textbf{(6)}$$

Where x^p is constituted by the features of Ω only.

$$s_k(x^p) = \mu_{ck}(x^p) \times \left(1 - \mu_{ck}(x^p)\right) \quad \textbf{(7)}$$ and

$$s_{k'k}(x^p) = \frac{1}{2} \times \left[\mu_{ck}(x^p) \times \left(1 - \mu_{ck'}(x^p)\right)\right] + \frac{1}{2} \times \left[\mu_{ck'}(x^p) \times \left(1 - \mu_{ck}(x^p)\right)\right] \quad \textbf{(8)}$$

$\mu_{ck}(x^p)$ and $\mu_{ck'}(x^p)$ are the membership values of the pattern x^p in classes C_k and $C_{k'}$, respectively. α_k is the normalizing constant for class C_k which takes care of the effect of relative sizes of the classes. Note that s_k is zero (minimum) if $\mu_{ck} = 1$ or 0, and is 0.25 (maximum) if $\mu_{ck} = 0.5$. On the other hand, $s_{k'k}$ is zero (minimum) when $\mu_{ck} = \mu_{ck'} = 1$ or 0, and is 0.5 (maximum) for $\mu_{ck} = 1$, $\mu_{ck'} = 0$ or vice versa.

Therefore, the term $s_k \Big/ \sum_{k \neq k'} s_{k'k}$, is minimum if $\mu_{ck} = 1$ and $\mu_{ck'} = 0$ for all $k \neq k'$ i.e., if the ambiguity in the belongingness of a pattern x^p to classes C_k and $C_{k'}$ is minimum (pattern belongs to only one class). It takes its maximum value when $\mu_{ck} = 0.5$ for all *k*. In other words, the value of **E** decreases as the belongingness of the patterns increases to only one class (i.e., compactness of individual classes increases) and at the same time decreases for other classes (i.e., separation between classes increases). **E** increases when the patterns tend to lie at the boundaries between classes (i.e. $\mu \rightarrow 0.5$). The objective in feature selection problem, therefore, is to select those features for which the value of ***E*** is minimum [15]. In

order to achieve this, the membership $\mu_{ck}(x^p)$ of a pattern x^p to a class is defined, with a multi-dimensional π - function which is given by

$$\mu_{ck}(x^p) \quad \begin{cases} = 1 - 2\, d_k^2(x^p) & \text{if } 0 \le d_k^2(x^p) < 0.5 \\ = 2\,[1 - d_k(x^p)]^2 & \text{if } 0.5 \le d_k^2(x^p) < 1 \\ = 0 & \text{otherwise} \end{cases} \quad \textbf{(9)}$$

The distance $d_k(x^p)$ of the pattern x^p from m_k (the center of class C_k) is defined as:

$$d_k(x^p) = \left[\sum_i \left(\frac{x_i^p - m_{ki}}{\lambda_{ki}} \right)^2 \right]^{1/2}, \quad \textbf{(10)} \qquad \text{where}$$

$$\lambda_{k_i} = 2 \max_p \left(\left| x_i^p - m_{ki} \right| \right) \quad \textbf{(11)}$$

$$\text{And } m_{ki} = \frac{\sum_{p \in C_k} x_i^p}{|C_k|} \quad \textbf{(12)}$$

Let us now explain the role of α_k. **E** is computed over all the samples in the feature space irrespective of the size of the classes. Therefore, it is expected that the contribution of a class of bigger size (i.e. with larger number of samples) will be more in the computation of **E**. As a result, the index value will be more biased by the bigger classes; which might affect the process of feature estimation. In order to overcome this i.e., to normalize this effect of the size of the classes, a factor α_k, corresponding to the class C_k, is introduced. In the present investigation, we have chosen $\alpha_k = 1/|C_k|$. However, other expressions like $\alpha_k = 1/prob_k$ or $\alpha_k = 1 - prob_k$ could also have been used.

If a particular subset (F_1) of features is more important than another subset (F_2) in characterizing / discriminating the classes / between classes then the value of ***E*** computed over F_1 will be less than that computed over F_2. In that case, both individual class compactness and between class separation would be more in the feature space constituted by F_1 than that of F_2. In the case of individual feature ranking (that fits to our need for feature estimation), the subset F contains only one feature [15].

Now, using feature estimation index we are able to calculate the *FWA* of each feature. As mentioned above, the smaller the value of ***E*** of a feature, the more significant that feature is. On the other hand, with *FWA* we mean that the larger its value for a given feature, the more significant that feature is. So we calculate the *FWA* of a feature this way: suppose $a_1, a_2, \ldots\ldots a_n$ are *n* features of a data set and *E* (a_i) and *FWA* (a_i) are feature estimation index and feature-weight assignment of feature a_i, respectively so

$$FWA(a_i) = \frac{\left(\sum_{j=1}^{n} E(a_j) \right) - E(a_i)}{\sum_{j=1}^{n} E(a_j)}, \qquad 1 \le i \le n \quad \textbf{(13)}$$

With this definition, *FWA* (a_i) is always in the interval [0.1]. So we define vector *FWA* which its *i*th element is *FWA* (a_i). Till now we have calculated *FWA* of each feature of the data set. Now we should take into account these values in calculating the distance between data points, which is of great significance in clustering.

ISBN: 978-972-8924-88-1 © 2009 IADIS

4. MODIFIED DISTANCE MEASURE FOR THE NEW *FCM* ALGORITHM

Two distance measures are used in *FCM* widely in literature: Euclidian and Mahalanobis distance measure. Suppose *x* and *y* are two pattern vectors (we have introduced pattern vector in section 3). The Euclidian distance between *x* and *y* is:

$$d^2(x, y) = (x - y)^T (x - y) \quad \textbf{(14)}$$

And the Mahalanobis distance between *x* and a center *t* (taking into account the variability and correlation of the data) is:

$$d^2(x, t, C) = (x - t)^T C^{-1} (x - t) \quad \textbf{(15)}$$

In Mahalanobis distance measure *C* is the co-variance matrix. Using co-variance matrix in Mahalanobis distance measure takes into account the variability and correlation of the data. To take into account the weight of the features in calculation of distance between two data points we suggest the use of $(x\text{-}y)_m$ (modified *(x-y)*) instead of *(x-y)* in distance measure, whether it is Euclidian or Mahalanobis. $(x\text{-}y)_m$ is a vector that its *i*th element is obtained by multiplication of *i*th element of vector $(x - y)$ and *i*th element of vector ***FWA***. So, with this modification, *equ.14* and *equ.15* will be modified to this form:

$$d_m^2(x, y) = (x - y)_m^t (x - y)_m \quad \textbf{(16)}$$ and

$$d_m^2(x, t, C) = (x - t)_m^t \; C^{-1} (x - t)_m \quad \textbf{(17)}$$ respectively , where

$$(x - y)_m(i) = (x - y)(i) \times FFWI \;\; (i) \quad \textbf{(18)}.$$

We will use this modified distance measure in our algorithm of clustering data set with different feature-weights in next section. To illustrate different aspects of the distance measures mentioned above let's look at some graphs in Fig.4 Points in all graphs are at equal distance (with different distance measures) to the center. A circumference in graph **A** represents points with equal Euclidian distance to the center. In graph **B**, points are of equal Mahalanobis distance to the center. Here the co-variance matrix is: $C = \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix}$ In this case the variable **Y** has more variability than the variable **X**, then, even if the values in the y-axis appear further from the origin with respect to the Euclidean Distance, they have the same Mahalanobis distance as those in the x-axis or the rest of the ellipsoid.

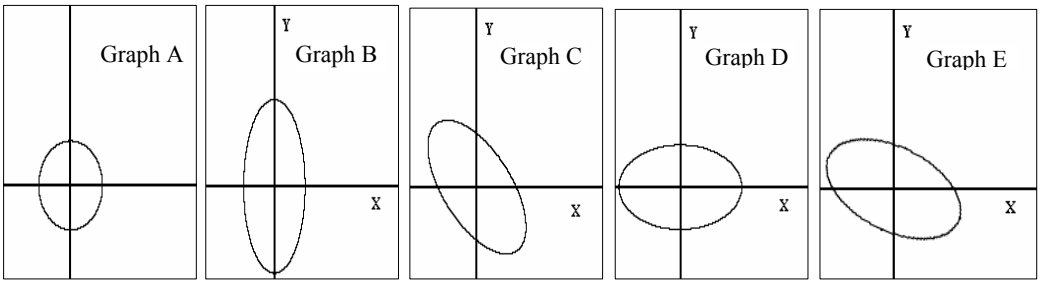


Figure 4. Point with Equal Distance to the Center

In the third case, let's assume that the parameters C is given by $C = \begin{pmatrix} 2.5 & -1.5 \\ -1.5 & 2.5 \end{pmatrix}$ Now the variables have a covariance different from zero. As a consequence, the ellipsoid rotates and the direction of the axis is given by the eigenvectors of **C**. In this case, greater values of **Y** are associated with smaller values of **X**. In other words, every time we move up, we also move to the left, so the axis given by the *y*-axis rotates to the left (see graph (**C**)). Graphs **D** and **E** demonstrate point with equal modified Euclidian and modified Mahalanobis distance to the centre, respectively. In both of them ***FWA*** vector is **FWA=** (0.33 0.67), and in graph **E**, **C** is equal to what it was in graph **C**. Comparing graphs **C** and **E**, we can conclude that in graph **E** in addition to variability and correlation of data, the *FWA* of features is considered in calculating distances.

5. NEW FEATURE WEIGHTED *FCM* ALGORITHM

In this section we propose the new clustering algorithm, which is based on *FCM* and extend the method that is proposed by [15] for determining *FWA* of features and, moreover, uses modified Mahalanobis measure of distance, which takes into account the *FWA* of features in addition to variability of data. As mentioned before, despite *FCM*, this algorithm clusters the data set based on weights of features. In the first step of this algorithm we should calculate the ***FWA*** vector using method proposed in [15]. To do so, we need some clusters over the data set to be able to calculate m_{k_i} and $d_k(x^p)$ (having these parameters in hand, we can easily calculate the feature estimation index for each feature. see section 3). To have these clusters we apply *FCM* algorithm with Euclidian distance on the data set. The created clusters help us to calculate the ***FWA*** vector. This step, in fact, is a pre-computing step. In the next and final step, we apply our Feature weighted *FCM* algorithm on the data set, but here we use modified Mahalanobis distance in *FCM* algorithm.
The result will be clusters which have two major difference with the clusters obtained in the first step. The first difference is that the Mahalanobis distance is used. It means that the variability and correlation of data is taken into account in calculating the clusters. The second difference, that is the main contribution of this investigation, is that features weight index has a great role in shaping the clusters.

6. CONCLUSIONS

In this paper, we have presented a new clustering algorithm based on fuzzy c-mean algorithm which is salient feature is that it clusters data set based on weighted features. We used a feature estimation index to obtain *FWA* of each feature. The index is defined based on the aggregated measure of compactness of the individual classes and the separation between the classes in terms of class membership functions. The index value decreases with the increase in both the compactness of individual classes and the separation between the classes. To calculate the feature estimation index we passed a pre-computing step which was a fuzzy clustering using *FCM* with Euclidian distance. Then we transformed the values into the *FWA* vector which its elements are in interval [0, 1] and each element shows the relative significance of its peer feature. Then, we merged the *FWA* vector and distance measures and used this modified distance measure in our algorithm. The result was a clustering on the data set in which weight of each feature plays a significant role in forming the shape of clusters.

ACKNOWLEDGEMENTS

This work is supported by FutureComm, the PRTLI project of Higher Education Authority (HEA), Ireland.

REFERENCES

1. Hall, L.O., Bensaid, A.M., Clarke, L.P., et al., 1992. *"A comparison of neural network and fuzzy clustering techniques in segmentation magnetic resonance images of the brain"*. IEEE Trans. Neural Networks 3.
2. Hung M, D. ang D, 2001 *"An efficient fuzzy c-means clustering algorithm"*. In Proc. the 2001 IEEE International Conference on Data Mining.
3. Han J., Kamber M., 2001 *"Datamining: Concepts and Techniques"*. Morgan Kaufmann Publishers, San Francisco.
4. Bezdek, J.C., 1981. *"Pattern Recognition with Fuzzy Objective Function Algorithms"*. Plenum, New York.
5. Dunn, J.C., 1974. *"Some recent investigations of a new fuzzy partition algorithm and its application to pattern classification problems"*. J. Cybernetics
6. Cannon, R.L., Dave, J., Bezdek, J.C., 1986. *"Efficient implementation of the fuzzy c means clustering algorithms"*. IEEE Trans. Pattern Anal. Machine Intell
7. Huang JZ , Ng MK , Rong H and Li Z.,2005. *"Automated Variable Weighting in k-Means Type Clustering"*. IEEE Transactions on Pattern Analysis & Machine Intelligence, Vol. 27, No. 5.

ISBN: 978-972-8924-88-1 © 2009 IADIS

8. Desarbo W.S., Carroll J.D.; Clark, and Green P.E., 1984 *"Synthesized Clustering: A Method for Amalgamating Clustering Bases with Differential Weighting Variables,"* Psychometrika, vol. 49.
9. Fisher, R., 1936. *"The use of multiple measurements in taxonomic problems"*. Ann. Eugenics 7.
10. Krishnapuram, R., Kim, J., 1999. *"A note on the Gustafson–Kessel and adaptive fuzzy clustering algorithms"*. IEEE Trans. Fuzzy Syst. 7.
11. Wu, K.L., Yang, M.S., 2002. *"Alternative c-means clustering algorithms"*. Pattern Recog. 35.
12. Zhao, S.Y., 1987. *"Calculus and Clustering"*. China Renming University Press.
13. Gustafson, D.E., Kessel, W., 1979. *"Fuzzy clustering with a fuzzy covariance matrix"*. In: Proceedings of IEEE Conference on Decision Control, San Diego, CA.
14. Hopner , K, R., Runkler, 1999 *"Fuzzy Cluster Analysis"*, John Wily & sons.
15. Pal S. K. and Pal A. (Eds.) 2002, *"Pattern Recognition: From Classical to Modern Approaches"*. World Scientific, Singapore.
16. de Oliveira J.V., Pedrycz W., 2007, *"Advances in Fuzzy Clustering and its Applications"*, John Wily & sons.
17. X. Wang, Y. Wang and L. Wang.,2004 "*Improving fuzzy c-means clustering based on feature-weight learning*", Pattern Recognition Letters **25**.

Appendix A: Distributed Clustering

Article II

Privacy Preserving Distributed Learning Clustering Of HealthCare Data Using Cryptography Protocols

Ahmed M. Elmisery, Huaiguo Fu

In Proceedings of the 34th IEEE Annual International Computer Software and Applications Workshops (COMPSACW 2010), Seoul, South Korea, July 2010.

2010 34th Annual IEEE Computer Software and Applications Conference Workshops

Privacy Preserving Distributed Learning Clustering Of HealthCare Data Using Cryptography Protocols

Ahmed M. Elmisery, Huaiguo Fu

Telecommunications Software & Systems Group
Waterford Institute of Technology, Waterford, Ireland
Waterford, Ireland

Abstract—**Data mining is the process of knowledge discovery in databases (centralized or distributed); it consists of different tasks associated with them different algorithms. Nowadays the scenario of one centralized database that maintains all the data is difficult to achieve due to different reasons including physical, geographical restrictions and size of the data itself. One approach to solve this problem is distributed databases where different parities have horizontal or vertical partitions of the data. The data is normally maintained by more than one organization, each of which aims at keeping its information stored in the databases private, thus, privacy-preserving techniques and protocols are designed to perform data mining on distributed data when privacy is highly concerned. Cluster analysis is a frequently used data mining task which aims at decomposing or partitioning a usually multivariate data set into groups such that the data objects in one group are the most similar to each other. It has an important role in different fields such as bio-informatics, marketing, machine learning, climate and healthcare. In this paper we introduce a novel clustering algorithm that was designed with the goal of enabling a privacy preserving version of it, along with sub-protocols for secure computations, to handle the clustering of vertically partitioned data among different healthcare data providers.**

Keywords-Clustering; privacy; Cryptography

1. INTRODUCTION.

With the advances in information and communication technology, information sharing for healthcare organizations became a vital requirement. However inappropriate sharing and usage of healthcare data could threaten patients' privacy[1].The healthcare providers need to share their data across different health organizations both within the country and with other countries that might have lesser privacy and security standards. Ideally, most health providers want to perform some statistical operations and extract knowledge from private database without revealing any additional information of each individual database.

Clustering is the process of assigning data points with similar properties to the same group and dissimilar data points to different groups [2]. Many clustering algorithms have been proposed in [3-9] based on different similarity or dissimilarity measurers. In this paper, we present an enhanced clustering algorithm called distributed local clustering (DLC); it was designed with the goal of enabling a privacy preserving version of the algorithm. Our experiments show that this algorithm produces clusters with acceptable accuracy with different shapes, sizes and densities of clusters. The private distributed local clustering (PDLC) based on a set of protocols to support distributed clustering over vertically partitioned private databases while minimizing the data breach among individual parties

2. DISTRIBUTED LOCAL CLUSTERING (DLC) ALGORITHM- AN OVERVIEW.

The DLC algorithm was designed by [3] to work in distributed environment, it requires only three parameters and it follows the following two steps:

- Local Learning and analysis step *(LLA).*
- Distributed clustering step *(DC).*

2.1 Local Learning and analysis Step (LLA)

LLA is elementary step, where the algorithm starts detecting dense regions and outliers from the data set in each local site using an influence function; that proposed in [6]. we used a gaussian influence function as an indicator, that is calculated for nearest neighbors only based on study in [8], All other points can be neglected without causing considerable error. Then we calculate the field function for a point as a summation of influence in its nearest neighbors. Detailed description for LLA listed as follows:

1. Influence of x on y (x, y ∈ F^d): $f^x_{Gauss}(y) = e^{-\frac{d(x,y)^2}{2\sigma^2}}$.
2. The field function for a point: $f^D_{Gauss}(y) = \sum_{i=1}^{k} e^{-\frac{d(x,y)^2}{2\sigma^2}}$ where *k* are the nearest neighbors for y.
3. Calculate outlierness degree factor (ODF) [8]: $ODF(s_i) = \frac{f^D(y)/k}{f^D(s)}$, if $ODF(s_i) \gg 1$, s_i is local outlier.

978-0-7695-4105-1/10 $26.00 © 2010 IEEE
DOI 10.1109/COMPSACW.2010.33

4. Based on local parameter α , $\forall f^D(y) \geq \alpha$, y is candidate local core point. Note that α value should be a point that shows variation in densities.
5. Based on influence and field functions it calculates initial cluster density [4] as follows:

$$D_{cluster_i} = {size_{cluster_i}}/{D_{Av}} \quad , \quad D_{Av} = {f^D(y)}/{k}$$

6. Check each data point if it will increase or decrease the density of the clusters when it joined or left clusters respectively within fixed threshold β, and then calculated:

$$D_{cluster_e} = {size_{cluster_e}}/{D_{Av}}, \; D_{Av} = {\sum_{i=1}^{n} f^x(y)}/{n}$$

7. If $D_{cluster_e} > D_{cluster_i}$
   The clusters that amplify density gain when data points joined it; are chosen as a candidate clusters from the density perspective, else keep the current.
8. Recalculate the influence and field functions to each $cluster_i$'s member; and the highest density point will be local core point.

2.2 Distributed Clustering Step (DC)

The learning and analysis step has a various amenities:

- Remove outliers and low density points from dataset.
- Select dense points as local core points.
- Speed the calculations for DC by building local table at each site contains densities.
- Limiting number of disclosed core points by limiting factor α that confines only a certain local core points (but α values in all sites should be in certain range in order to achieve good accuracy).

The DC algorithm uses the single link (slink) algorithm in [9], but with some modification to estimate merge error based on [10] and uses membership function in [11] with modification based on the field function notation. The algorithm takes one input which is the least error threshold *(LET)*. Figure 1 outlines the modified algorithm:

Algorithm : DC

1. Calculate the summation of all field functions for each point in all sites and store it in sorted table.
2. All sites agree on common α range to classify high density points.
3. Select the highest density point to be the starting point for current cluster.
4. **If** the same point in all sites ⊂ dense point's cluster neighbors
   then Expand the current cluster by adding this point
   Else Assign each point to its nearest dense point according to:

$$m(y_j|x_i) = \frac{\|x_i - y_j\|^2}{\sum_{j=1}^{k}\|x_i - y_j\|^2}$$

5. Start a new cluster and repeat steps 2 to step 4 until all points are clustered
6. Assign outliers based on step 4. To the nearest cluster.
7. according to [10] merging two clusters based on least error:
   $error\ (y_i \cup y_{i+1}) = \frac{y_i.w_i \times y_{i+1}.w_{i+1} \times d(y_i, y_{i+1})^2}{y_i.w_i + y_{i+1}.w_{i+1}}$
   the membership function calculated as fellow:
   $w_i(y_i|x_1, x_2, ..x_o) = \sum_{n=1}^{o} \frac{\| f^{x_n}(y_i)\|^2}{\sum_{j=1}^{k}\|x_i - y_j\|^2}$
8. Repeat step 7, until the error > LET.

Figure 1: DC algorithm.

The algorithm starts by calculating total density for each point across different sites relying on *LLA* findings; then it selects the highest density point as starting point for current cluster and expands this cluster by adding these common points in all sites that exist within this dense point's cluster. In case of a different point introduced it calculates the membership for that point to be element of this cluster, then it forms a new cluster and repeats the previous steps. After clustering all the points, the algorithm starts to investigate merging clusters with each other as long as this achieves less error than *LET*. For more information about DLC , the reader is referred to [3].

Time complexity: if we assume that m is the cardinality of Database; the algorithm gets nearest neighbors in $O(m \log m)$ time, calculates the density for each point depend of its o-neighbors in $O(om \log om)$, gets highest density points $O(m \log m)$,removing low density points requires $O(w)$ where $w = m - p$, and the clustering process takes $O(m + m \log m)$ where $p = m - a$ (a is number of high dense points). The overall complexity $O(3m \log m + om \log om + w + m)$ which is good performance in distributed environment.

Communication complexity: all the required operations are done using distributed summation which has linear communication overhead.

2.3 Experimental results

We used synthetic data set to test our algorithm; the data set contains 13 clusters of different shape, size and density; the data set includes a noise at the level of 6%. The DLC identifies all 13 clusters correctly, the resulting clusters shown in the figure 2. The algorithm was coded in C and executed on a Dell precision running Windows XP with 2GB RAM and Core duo CPU 2.4 GHz. *LLA* detects different dense points in the data set and produces 32 sub-

cluster. Then initial *DC* step was performed in these sub-clusters and it forms 18 cluster, using the merge function in *DC* reduce number of clusters to 13.

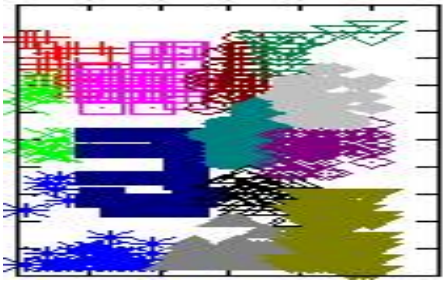


Figure 2: The Final Clusters for DLC Algorithm

3. Private Distributed Learning Clustering (PDLC)

In the following sections, we will present a privacy preserving version of DLC for vertically partitioned data.

3.1 Notations

n: total number of parties, we assume $n \geq 3$.
p_i: site i.
m_i: number of records in site i.
D_i: private portion of the database D for site i.
TID: join key between distributed data portions.
K: total number of clusters.
y_i: local core point in site i
x_i: any other point at site i (noncore point)

3.2 Problem formulation

There is a distributed database D in n ($n \geq 3$) sites, D has *M-dimensions*. The objective is to cluster D to K clusters based on the previous algorithm without disclosing any additional information, Each site j has D_j with a set of fields C_j , where $M = \sum_{j=1}^{N} |C_j|$, $C_i \cap C_j = \emptyset \ \forall \ i \neq j$. There is a join key in all D_j $(2 \leq j \leq N)$ called *TID*. Several additional concepts are required to describe PDLC to cluster vertically partitioned database.

Definition 1: The portion of point y at site j denoted by y_j is a record has set of fields c_j where the values of these fields are the same with y.

Definition 2: A point y_j at site j is said to be local dense point; if its field function $f^D(y_j) \geq \alpha$, where α is local parameter for site j .

One simple scenario is to run *LLA* step in each site to detect dense points without a *DC* step, and then use secure set intersection protocol to get common ones. That achieves good privacy but gives inaccurate results due to:

- Ignore the dependency influence of different fields in all the sites.
- Requires the same value for α because different values affect the accuracy of the results.

3.3 Cryptographic Primitives

In this section, we will detail a set of cryptographic tools used in *PDLC*:-

3.3.1. Basic assumption

This work is based on secure multi-party computation (SMC), generally speaking SMC deals with privacy concerns in distributed environment while ensuring correctness of the computation and that no more information is revealed to a participant in the computation than can be inferred from that participant's input and output [12]. SMC assumes that the communication channels between participant sites are secure, that is each pair of parties is connected by a reliable and private channel, also we assume all participant sites are semi-honest (which is the case in reality, where different partners need to accomplish some goals by the output mining model). A secure protocol for semi-honest model can be transformed into a protocol for malicious model using zero knowledge proof.

3.3.2. Homomorphic encryption

Homomorphic encryption schema [13] is an encryption schema which allows certain algebraic operations to be carried out on the encrypted plain text, by applying different operation to the corresponding cipher text. Let $(pub_i, priv_i)$ denote a cryptographic public/private key pair of site i, and $ENC_{pub_i}(.)$ denotes the encryption function with public key pub_i , $DEC_{priv_i}(.)$ denotes the decryption function with private key $priv_i$. A homomorphic encryption is commutative encryption if the following three equations hold:

- For any given feasible public keys $\text{pub}_1, \text{pub}_2, \text{pub}_3, \ldots\ldots, \text{pub}_n$, any t in plain text domain T, and any permutations of i, j , $\text{ENC}_{\text{pub}_{i_1}}\left(\ldots \text{ENC}_{\text{pub}_{i_n}}(\text{t}).\right) = \text{ENC}_{\text{pub}_{j_1}}\left(\text{ENC}_{\text{pub}_{j_n}}(\text{t})..\right)$.
- $\text{ENC}_{\text{pub}_{i_1}}(\text{t}_1) \times \text{ENC}_{\text{pub}_{i_1}}(\text{t}_2) = \text{ENC}_{\text{pub}_{i_1}}(\text{t}_1 + \text{t}_2)$.
- $\text{ENC}_{\text{pub}_{i_1}}(\text{t}_1)^{\text{v}} = \text{ENC}_{\text{pub}_{i_1}}(\text{vt}_1)$.

3.3.3. Secure Distributed Summation (SDS)

SDS can be described as follows: there are $n_i (i > 3)$ sites, and each site has private input v_i. All the sites want to compute $\sum_{i=1}^{n} v_i$, without revealing the private input of each site to other sites. The sum is sometimes shared by two or more parties in order to do further computation. modified version of SDS can be found in [14] , we used SDS as sub-protocol with some modifications.

- p_i's ($2 \le i \le n$) randomly select a key generator site : p_1 (changed every run)
- p_1 generates a cryptographic key pair $(pub_1, priv_1)$ of a secure homomorphic encryption, and publish its public key pub_1 to all sites.
- p_1 generates random noise RN_m, encrypts the point's densities in each cluster , computes $ENC_{pub_1}(v_{11} + RN_{11}), TID_1$ ……+ $ENC_{pub_1}(v_{m1} + RN_{m1}), TID_m$ and sends them to p_2.
- p_2 checks the $TIDs$, adds its encrypted values to those match his own and appends the list with new TIDs (if any) . p_2 computes $ENC_{pub_1}(v_{m1} + RN_{m1}) \times ENC_{pub_1}(v_{m2} + RN_{m2}), TID_m$=$ENC_{pub_1}((v_{m1} + RN_{m1}) + (v_{m2} + RN_{m2})), TID_m$ and send them to p_i ($3 \le i \le n$).
- The protocol progress until reaching to p_n which will have $ENC_{pub_1}\left(\sum_{i=1}^{n}\sum_{j=1}^{m} v_{ji} + \sum_{i=1}^{n}\sum_{j=1}^{m} RN_{ji}\right)$
- In the same way, p_{n-1} computes $ENC_{pub_1}\left(\sum_{i=1}^{n}\sum_{j=1}^{m} RN_{ji}\right)$ only.
- p_n invokes to p_1 to get the decryption key. p_n sends part of encrypted points' densities to p_{n-1} with the decryption key.
- p_{n-1} removes noise by calculating $ENC_{pub_1}\left(\sum_{i=1}^{n}\sum_{j=1}^{m} RN_{ji}\right)^{-1}$, then adds this to the received points after matching it with $TIDs$ as following:$ENC_{pub_1}\left(\sum_{i=1}^{p}\sum_{j=1}^{m} v_{ji} + \sum_{i=1}^{p}\sum_{j=1}^{m} RN_{ji} - \sum_{i=1}^{p}\sum_{j=1}^{m} RN_{ji}\right), TID_m$ then it decrypt the results And sends non matching $TIDs$ to p_n
- In the same way p_n first adds the received summed noise after matching their TIDs to its data to remove the noise effect and then it decrypts them.
- Both p_{n-1} and p_n sort points' densities with their corresponding $TIDs$ in table and broadcast it to all sites.

Security Analysis: if all parties fellow the protocol, they will get sorted global list of densities for their points. In this sub-protocol, we rely on some properties of homomorphic encryption that listed in the previous section. We assume p_1 , p_n , p_{n-1} not colluding sites because they don't know which one will be selected. Also there are two levels of privacy preserving for each site before sending the data;

- Specific random noise added to each density value that is only known to the owner site.
- Encryption of previous values using public key of selected random site.

Due to that, other sites can't breach individual sites values.

Time Complexity Analysis: $O(3im^2 + 3im + 4im\ log\ im)$ where T is the size of *TIDs*, i is number of sites and m is cardinality of data in each site.

Communication Complexity Analysis: $O(T(3im + 2pm + i + 2))$ where p is the number of elements send for p_n sends to p_{n-1}.

3.3.4. Secure division Protocol (SDC)

There are n parties, each one has two values v_i and y_i ; they want to securely compute $\frac{\sum_{i=1}^{n} v_i}{\sum_{i=1}^{n} y_i}$. So by using secure multi-party addition they can separately compute v_i's and y_i's, then one site say p_1 receives $r_i = \frac{v_i}{y_i}$ ($2 \le i \le n$) from other sites, computes $\sum_{i=1}^{n} r_i$ which equal to division of two values.

3.3.5. Secure Intersection Protocol (SIC)

Based on the idea In [15],[16], we extend this protocol to get the intersection of common members in each cluster from different sites without revealing any additional information except their cardinalities and *TIDs*.

- Assume the after running SDS, the most density point is y_i.
- p_i's ($2 \le i \le n$) randomly select a key generator site : p_1 (changed every run)
- p_1 generates a cryptographic key pair $(pub_1, priv_1)$ of a secure homomorphic encryption and publish its public key pub_1 to p_2.
- p_2 generates $(pub_2, priv_2)$ and publish its public key pub_2 to p_i ($3 \le i \le n$)
- p_i randomly permutes $TIDs_i$ of the cluster neighbors of point y_i to $\widetilde{TIDs_\iota}$ and encrypts $ENC_{pub_2}(\widetilde{TIDs_\iota})$ with the public key pub_2 and sends it to p_1.
- After receiving all shares, p_1 computes $ENC_{pub_2}(A_i)$ and $ENC_{pub_2}(B_i)$; where A_i is the set of all common *TIDs* and B_i is the set different *TIDs*; then it sends $ENC_{pub_1}(ENC_{pub_2}(A_i))$, $ENC_{pub_1}(ENC_{pub_2}(B_i))$ to p_2.
- p_2 decrypts the two sets, and intersects its shares with $ENC_{pub_1}(A_i)$, $ENC_{pub_1}(B_i)$ and sends the results to p_1.
- p_1 decrypts the two sets and broadcast A_i, B_i to all sites. Each site removes its permutation and builds a table for the clusters and densities in other sites.

Security Analysis: the homomorphic encryption is deterministic algorithm; there is unique cipher text for each plain text. Therefore A_i=$ENC_{pub_2}(\widetilde{\text{TIDs}_3}) \cap ENC_{pub_2}(\widetilde{\text{TIDs}_4}) \dots \dots \dots \dots \cap ENC_{pub_2}(\widetilde{\text{TIDs}_\iota})$, the cipher text which is not the same will stored in B_i. Only p_1, p_2 get all *TIDs*. But they are not able to figure out anything because *TIDs* are permutated with parameter only know to owner site but not p_1 and p_2.

Time Complexity Analysis: $O(Ti \log Ti)$ where T is the size of *TIDs*, and *i* is number of sites.

Communication Complexity Analysis: is $O(Ti)$.

4. PDLC FOR DISTRIBUTED HEALTHCARE DATA SCENARIO

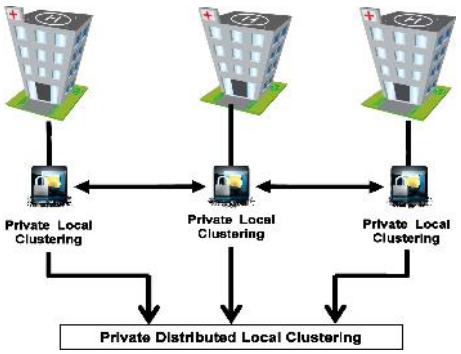


Figure 3: Simple scenario for PDLC

Here we will give a simple scenario to use *PDLC* in clustering distributed data. According to the "Health insurance portability and accountability act" (HIPAA), public bodies, such as hospitals and universities, are not allowed to disclose unauthorized personal information. Now suppose several hospitals want to gain some knowledge, helping them to quickly diagnose the arriving patients with some specific conditions, without disclosure of their own patients' information. Therefore, they have to collaboratively cluster their patients' data to get that knowledge. A framework for this problem where *PDLC* was employed is shown in Figure 3. All the Hospitals first run *LLA* step in its local private data to identify local dense points then *LLA* starts forming clusters based on the density gain. The *LLA* changes the way we look to the data, instead of acting with data values directly, it uses density based on local neighbors values that is very efficient to identify arbitrary shape clusters of different sizes and densities also it helps the *DC* step to form a global clusters based on these values. Next, they execute the *PDC* protocol that is a private version of *DC* see figure 4. Note that all the computations are done in a finite field of size N.

Algorithm : PDC

1. p_1 Invokes to run SDS sub-protocol to get the summation of all field functions for each point in the data set.
2. p_n , p_{n-1} publish a sorted list of point's densities. Then all sites agree on common α range.
3. Select the highest density point to be the starting point for current cluster.
4. The sites invoke SIC sub-protocol to get common points in different sites' clusters
5. Expand the current cluster by adding these common points.
6. repeat
   Starting with each point in B_i , Based on the published densities, p_1 invokes SDS and each site calculates its share of the following equations: $\frac{\|x_i-y_j\|^2}{\sum_{j=1}^{k}\|x_i-y_j\|^2}$, and send it to the site p_{n+1}.
   Until no points in B_i.
7. p_n adds its share to the equations and invokes SDC to calculate : $m(y_j|x_i) = \sum_{i=1}^{n} \frac{\|x_i-y_j\|^2}{\sum_{j=1}^{k}\|x_i-y_j\|^2}$ where k is the top k highest density points determined by α , then it assigns the low density point to the nearest dense point and stores these values.
8. Start a new cluster based on the next highest density point and repeat steps 4 to step 7 until all points are clustered
9. Assign outliers based on step 7. To the nearest cluster.
10. p_n Builds a table based on step 7 and calculates $w_i(y_i|x_1, x_2, .. x_o) = \sum_{n=1}^{o} \frac{\| f^{x_n}(y_i)\|^2}{\sum_{j=1}^{k}\|x_i-y_j\|^2}$ where o is the number of elements in each new cluster, then it broadcast this table to all sites.
11. Based on published densities and memberships p_1 calculates least error for merging two clusters $error\ (y_i \cup y_{i+1}) = \frac{y_i.w_i \times y_{i+1}.w_{i+1} \times d(y_i, y_{i+1})^2}{y_i.w_i + y_{i+1}.w_{i+1}}$
12. Repeat step 11, until the error> LET.

Figure 4: PDC algorithm

4.1 A worst case

If p_n , p_{n-1} , p_1 are colluding sites. No site could know the results of *LLA* rather than the owner site because this is done locally and only the published information is the densities of these points from different data set attributes that change from site to site (Not the real values). The *DC* step works on the published densities only. Based on these information one site values cannot disclosed unless $n-4$ sites collude with p_n , p_{n-1} , p_1 , this assumption is not realistic in real life, and even so, values will be

estimations to the real ones with certain error degree due to σ values in influence function calculations.

5. Related Work

The problem was introduced first by [17], in this work the authors aims to achieve privacy preserving classification, they used the oblivious transfer protocol. In the same time [18] presented another solution to the problem based on randomization approach to achieve privacy preserving classification using decision trees. More work done to enhance the randomization approach and proposed in [19], [20]. researchers proposed enhanced solutions in [15], [14] using secure multiparty computations to perform association rule mining and privacy in k-means clustering for vertically partitioned data. More enhanced work in [21] proposes a privacy preserving k-means over arbitrarily partitioned data. A similar approach to ours can be found in [5] but they perform clustering using sampling density estimates and solve the problem using different approach.

6. Conclusion and Future wok

In this paper, we present a novel clustering algorithm for vertically partitioned data; we test the performance of that algorithm based on experiments and complexity analysis. Later we presented a private version of this protocol using protocols based on homomorphic encryption. Our protocol is robust against colluding attack.

We need to perform extensive experiments in real data set from UCI repository and compare the performance of *DLC* with other clustering algorithms, also we need to consider different data partitioning techniques, identify potential threats and add some protocols to ensure the privacy of the data against these threats. We also need to consider some powerful adversaries in parties, one way solve this problem is by using zero knowledge proof protocols; also we need to reduce the time and communication complexity for *LLA* and *DC* in order to be more efficient for large and dimensional data.

7. Acknowledgment

This work has received support from the Higher Education Authority in Ireland under the PRTLI Cycle 4 programme, in the project Serving Society: Management of Future Communications Networks and Services.

Special thanks to Mohamed Gaber from the School of Computing, University of Portsmouth, United Kingdom, for his valuable comments and feedback.

References

[1] M. Ashley Katz and K. Johnson. 29 Mar). Privacy Cases in HealthCare. Available: www.patientprivacyrights.org

[2] J. Han and M. Kamber, Data Mining: Concepts and Techniques (The Morgan Kaufmann Series in Data Management Systems): Morgan Kaufmann, 2000.

[3] A. Elmesiry, "Distributed Local Clustering (DLC) for Large Distributed Databases," WIT2010.

[4] Z. S. Ammar and M. M. Gaber, "DDG-Clustering: A Novel Technique for Highly Accurate Results," in IADIS European Conference on Data Mining (ECDM 2009), Algarve, Portugal, 2009.

[5] M. Klusch, et al., "Distributed clustering based on sampling local density estimates," presented at the Proceedings of the 18th international joint conference on Artificial intelligence, Acapulco, Mexico, 2003.

[6] A. Hinneburg, et al., "An efficient approach to clustering in large multimedia databases with noise," presented at the Int. Conf. on Knowledge Discovery in Databases, 1998.

[7] H. Fu and A. M. Elmisery, "A New Feature Weighted Fuzzy C-means Clustering Algorithm," in IADIS European Conference on Data Mining (ECDM 2009), Algarve, Portugal, 2009.

[8] J. Wen, et al., "Ranking outliers using symmetric neighborhood relationship," ed: Springer, 2006.

[9] S. Johnson, "Hierarchical clustering schemes," Psychometrika, vol. 32, pp. 241-254, 1967.

[10] Jr, "Hierarchical Grouping to Optimize an Objective Function," Journal of the American Statistical Association, vol. 58, pp. 236-244, 1963.

[11] A. Kalton, et al., "Generalized clustering, supervised learning, and data assignment," presented at the Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, San Francisco, California, 2001.

[12] U. Maurer, "Secure multi-party computation made simple," Discrete Appl. Math., vol. 154, pp. 370-381, 2006.

[13] A. J. Menezes, et al., Handbook of Applied Cryptography: CRC Press, Inc., 1996.

[14] J. Vaidya and C. Clifton, "Privacy-preserving k-means clustering over vertically partitioned data," presented at the Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, Washington, D.C., 2003.

[15] J. Vaidya and C. Clifton, "Privacy preserving association rule mining in vertically partitioned data," presented at the Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, Edmonton, Alberta, Canada, 2002.

[16] J. Vaidya and C. Clifton, "Secure set intersection cardinality with application to association rule mining," J. Comput. Secur., vol. 13, pp. 593-622, 2005.

[17] Y. Lindell and B. Pinkas, "Privacy Preserving Data Mining," presented at the Proceedings of the 20th Annual International Cryptology Conference on Advances in Cryptology, 2000.

[18] R. Agrawal and R. Srikant, "Privacy-preserving data mining," SIGMOD Rec., vol. 29, pp. 439-450, 2000.

[19] W. Du and Z. Zhan, "Using randomized response techniques for privacy-preserving data mining," presented at the Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining, Washington, D.C., 2003.

[20] H. Kargupta, et al., "Random-data perturbation techniques and privacy-preserving data mining," Knowl. Inf. Syst., vol. 7, pp. 387-414, 2005.

[21] G. Jagannathan and R. N. Wright, "Privacy-preserving distributed k-means clustering over arbitrarily partitioned data," presented at the Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, Chicago, Illinois, USA, 2005.

Appendix B: IPTV Recommender Service Scenario

Article III

Agent Based Middleware for Maintaining User Privacy in IPTV Recommender Services

Ahmed M. Elmisery, Dmitri Botvich

In Proceedings of the 3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems (MOBISEC 2011), Aalborg, Denmark, May 2011.

Agent Based Middleware for Maintaining User Privacy in IPTV Recommender Services

Ahmed M. Elmisery and Dmitri Botvich

Telecommunications Software & Systems Group
Waterford Institute of Technology, Waterford, Ireland

Abstract. Recommender services that are currently used by IPTV providers help customers to find suitable content according to their preferences and increase overall content sales. Such systems provide competitive advantage over other IPTV providers and improve the overall performance of the current systems by building up an overlay that increases content availability, prioritization and distribution that is based on users' interests. Current implementations are mostly centralized recommender service (CRS) where the information about the users' profiles is stored in a single server. This type of design poses a severe privacy hazard, since the users' profiles are fully under the control of the CRS and the users have to fully trust the CRS to keep their profiles private. In this paper, we present our approach to build a private centralized recommender service (PCRS) using collaborative filtering techniques and an agent based middleware for private recommendations (AMPR). The AMPR ensures user profile privacy in the recommendation process. We introduce two obfuscation algorithms embedded in the AMPR that protect users' profile privacy as well as preserve the aggregates in the dataset in order to maximize the usability of information for accurate recommendations. Using these algorithms provides the user complete control on the privacy of his personal profile. We also provide an IPTV network scenario that uses AMPR and its evaluations.

Keywords: Privacy, Clustering, IPTV Networks, Recommender System, Multi-Agent Systems.

1 Introduction

Internet protocol television (IPTV) is one of the most fast growing services in ICT; it broadcasts multimedia content in digital format via broadband internet networks using IP packet switched network infrastructure. Differently from conventional television, IPTV allows an interactive navigation of the available items [1]. IPTV providers employ automated recommender services by collecting information about user preferences for different items to create a user profile. The preferences of a user in the past can help the recommender service to predict other items that might be interested for him in the future.

Collaborative filtering (CF) technique is utilized for recommendation purposes as one of the main tools for recommender systems. CF is based on the assumption that

R. Prasad et al. (Eds.): MOBISEC 2011, LNICST 94, pp. 64–75, 2012.

people with similar tastes prefer the same items. In order to generate recommendations, CF cluster users with the highest similarity in their interests, then dynamic recommendations are then served to them as a function of aggregate cluster interests. Thus, the more the users reveal information about their preferences, the more accurate recommendations provided to them. However at the same time the more information is revealed to the recommender service about the user profile, the lower user privacy levels can be guaranteed. This trade-off acts as a requirement when designing a recommender service using CF technique. Privacy aware users refrain from providing accurate information because of their fears of personal safety and the lack of laws that govern the use and distribution of these data. Most service providers would try their best to keep the privacy of their users. But occasionally, when they are facing bankruptcy, they might sell it to third parties in exchange of financial benefits. In the other side, many service providers might violate users' privacy for their own commercial benefits. Based on a survey results in [2, 3] the users might leave a service provider because of privacy concerns. The information collected by recommender service breaches the privacy of the users in two levels.

1. The real identity of the user is available to a central server. That server can associate the user profile which contains his private information to his real identity. This is an obvious privacy breach, considering that a user does not want to reveal the link between his real identity and his profile, yet he wants to use the service in that server.
2. If the user is not known to the server, the server can try to de-anonymize the user identity by correlating the information contained in the user profile and some information obtained from other databases [4].

In this paper we proposed an agent based middleware for private recommendation (AMPR) that bear in mind privacy issues related to the utilization of collaborative filtering technique in recommender service and allow sharing data among different users in the network. We also present two obfuscation algorithms that protect the user privacy and preserve the aggregates in the dataset to maximize the usability of information in order to get accurate recommendations. Using these algorithms, gives the user a complete control on his personal profile, so he can make sure that the data does not leaves his side until it is properly desensitized. In the rest of this paper we will generically refer to news programs, movies and video on demand contents as Items. Section 2 describes some related work. In Section 3 we introduce our private centralized recommender service scenario in IPTV network. In Section 4 we introduce the proposed obfuscation algorithms used in our framework. Section 5 describes some experiments and results based on obfuscation algorithms for IPTV network. Section 6 includes the conclusion and future work.

2 Related Work

The majority of the literature addresses the problem of privacy for recommender services based on collaborative filtering technique, Due to it is a potential source of leakage of private information shared by the users as shown in [5]. In [6] it is

proposed a theoretical framework to preserve the privacy of customers and the commercial interests of merchants. Their system is a hybrid recommender that uses secure two party protocols and public key infrastructure to achieve the desired goals. In [7, 8] it is proposed a privacy preserving approach based on peer to peer techniques using users' communities, where the community will have a aggregate user profile representing the group as whole and not individual users. Personal information will be encrypted and the communication will be between individual users and not servers. Thus, the recommendations will be generated at client side. In [9, 10] it is suggest another method for privacy preserving on centralized recommender systems by adding uncertainty to the data by using a randomized perturbation technique while attempting to make sure that necessary statistical aggregates such as mean don't get disturbed much. Hence, the server has no knowledge about true values of individual rating profiles for each user. They demonstrate that this method does not decrease essentially the obtained accuracy of the results. Recent research work [11, 12] pointed out that these techniques don't provide levels of privacy as it was previously thought. In [12] it is pointed out that arbitrary randomization is not safe because it is easy to breach the privacy protection it offers. They proposed a random matrix based spectral filtering techniques to recover the original data from perturbed data. Their experiments revealed that in many cases random perturbation techniques preserve very little privacy. Similar limitations were detailed in [11].

3 Problem Formulation

3.1 System Model

We consider a system where PCRS is implemented as a third-party service that makes recommendations by consolidating the profiles received from multiple users. Each user has a set top box (STB) that stores his profile and host AMPR at his side. As shown in fig 1, the parties involved are the users, and the PCRS. We assume that PCRS follow the semi-honest adversary model, which is realistic assumption because the PCRS provider needs to accomplish some business goals and increase his revenues. Moreover, we assume the communication links between parties are secured by existing techniques. An IPTV provider uses this business model to reduce the required computational power, expenses or expertise to maintain an internal recommender service.

3.2 Design Goals

There are two requirements should be satisfied in the previous system model:

- IPTV providers care about the privacy of their catalogue which is considered an asset for their business. In the meantime they are willing to offer real users' ratings for different masked items to offer better recommendations for their users and increase their revenues.
- In the other side, privacy aware users worry about the privacy of their profiles, as sending their real ratings harm their privacy.

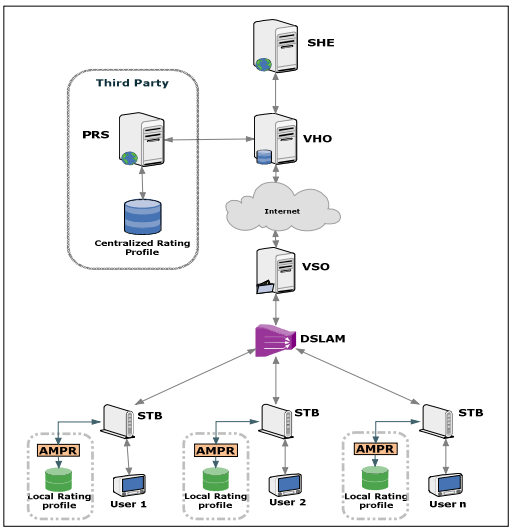


Fig. 1. Illustration of proposed combined IPTV Network

The AMPR employs two obfuscation algorithms that provide the users the required privacy level before submitting the profiles to the PCRS. Note that, we alleviate the user identity problems by using anonymous pseudonyms identities for users.

3.3 Threat Model

In this paper, AMPR provides a defence mechanism against the threat model proposed in [13] where the attacker colludes with some users inside the network to obtain some partial information about the process used to obfuscate the data and/or some of the original data items themselves. The attacker can then use this partial information for the reverse engineering of the entire data set.

4 Solution

In the next sections, we will present our proposed framework for preseving the privacy of customers' profiles show in fig 2.

4.1 PCRS Components

As show in fig 2, PCRS maintains a set data stores. The first data store is the masked catalogue of items that have been hashed using IPTV provider key or a group key. The second data store is the obfuscated users' profiles which contain users' pseudonyms and their obfuscated ratings and finally a peer cache which is an updated database about peers participated in previous recommendations formulation. The peer cache is updated from peer list database at client side. The PCRS communicates with the user through a manager unit. Finally, the clustering manager is the entity responsible for building recommendations model based upon the obfuscated ratings database.

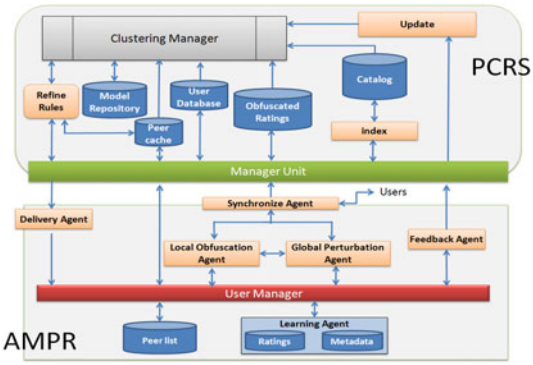


Fig. 2. PCRS framework

4.2 AMPR Components

The AMPR in the user side consists of different co-operative agents. Learning agent captures user preferences about items explicitly or implicitly to build a rating table and meta-data table. The local obfuscation agent implements CBT obfuscation algorithm to achieve user privacy while sharing the data with other users or the system. The global perturbation agent executes G-algorithm on the locally obfuscated collected profiles. These algorithms act as wrappers that obfuscate items' ratings before they are fed into the PCRS. Since the database is dynamic in nature, the local obfuscation agent desensitizes the updated data periodically, then synchronize agent send it to other users and PCRS. So the recommendations are made on the most recent ratings. More details about the recommendation process described in the next sub-section.

4.3 The Recommendation Process

The recommendation process based on the two stage obfuscation algorithms can be summarized as following more details can be found in [14]. The target user broadcasts message to other users in the IPTV network to request starting the recommendations process or update their centralized rating profiles stored at PCRS. The individual users who decided to participate in that process use the local obfuscation agent to perform *CBT* algorithm of their local rating profiles. They agree on same parameters, and then they submit their locally obfuscated profiles to the requester. The target user instructs his obfuscation agent to start *G* algorithm on the collected locally obfuscated profiles. After finishing the previous step, the target user submits all profiles to PCRS in order to receive recommendations.

5 Proposed Algorithms

In the next sub-sections, we provide two different algorithms that used by our agents to obfuscate the user profile in a way that secure his ratings in the un-trusted PCRS with minimum loss of accuracy. In our framework, each user has two datasets representing his/her profile. First one is the local rating profile which is perturbed

before merging it with similar users' profiles that rare willing to collaborate with him as part of the recommendation process. The second one is the centralized rating profile which is the output of the two obfuscation algorithms where the user can get recommendation directly from the PCRS based on it. We perform experiments on real datasets to illustrate the applicability of our algorithms and the privacy and accuracy levels achieved using them.

5.1 Local Obfuscation Using CBT Algorithm

We propose a new obfuscation algorithm called clustering based transformation (CBT) that have been designed especially for the sparse data problem in user profile. It is inspired from the block cipher idea in [15]. We present a new technique to build a transformation lookup table (TLUT) using clustering technique then approximate each point in the data set to the nearest representative value in the TLUT (the core-point for the cluster it belong to) with the help of similarity measures. The output of our obfuscation algorithm should satisfy two requirements:

1. Reconstructing the original data from the obfuscated data should be difficult, in order to preserve privacy.
2. Preserve the similarity between data to achieve accurate results.

We use local learning analysis (*LLA*) clustering method proposed in [16] to create the *TLUT*. It is important to attain an optimized *TLUT* because the quality of the *TLUT* obviously affects the performance of the transformation. *LLA* builds an initial *TLUT* and repeats the iteration till two conditions satisfied:

1. The distance function $d(x, c_i)$ between a point x and its corresponding value (core-point) c_i is minimized.
2. The distortion function between each dataset and its nearest value (core-point) becomes smaller than a given threshold.

CBT algorithm consists of following steps:

1. The user ratings stored as dataset D of c rows, where each row is sequence of fields $X = x_1\ x_2\ x_3 \ldots\ldots x_m$.
2. User ratings dataset D is portioned into $D_1\ D_2\ D_3 \ldots\ldots D_n$ datasets of length L, if total number of attributes in original is not perfectly divisible by L then extra attributes is added with zero value which does not affect the result and later it is removed at step 5.
3. Generate *TLUT* using *LLA* algorithm, *LLA* takes Gaussian Influence function as the similarity measure. Influence function between two data points x_i and x_j is given as

$$f_{Gauss}^{x_i}(x_j) = e^{-\frac{d(x_i, x_j)^2}{2\sigma^2}} \quad (1)$$

While the field function for a candidate core-point given by:

$$f_{Gauss}^{D}(x_j) = \sum_{s=1}^{k} e^{-\frac{d(x_j, x_{is})^2}{2\sigma^2}} \quad (2)$$

Clustering is performed on each dataset D_i, resulting to k clusters $C_{i1}, C_{i2}, C_{i3}, ..., C_{ik}$ and each cluster is represented by its core-points, i.e. core-point of j^{th} cluster of i^{th} dataset is $(C_{ij}) = \{c_1, c_2, c_3, ...c_L\}$ .Every single row portion falls in exactly one cluster. And The TLUT = (core-point (C_{i1}) , core-point(C_{i2}) , core-point(C_{i3}) ..,core-point (C_{ik}))

4. Each dataset D_i is transformed into new dataset D_i' using generated TLUT, each portion $Y_i = x_{(i-1)L+1}\ x_{(i-1)L+2}\ x_{(i-1)L+3} \ldots\ldots x_{iL}$ replaced by the nearest cluster core-point Z_i = core-point (C_{ij}) in which it falls.

$$Y_i \xrightarrow{transofrmed} Z_i$$

5. The transformation function is: $T(Y_i) = \{core - point(C_j \leftrightarrow d(Y_i, core - point(C_j) < d(Y_i, core - point(C_Z)\ \forall Z\}$
6. Now all the n transformed portions of each point are joined in the same sequence as portioned in step 2 to form a new k dimension transformed row data which replaces the X in the original dataset. In this way perturbed dataset D_i' is formed from original dataset D
7. Compute the privacy level by calculating the difference between the original dataset and transformed dataset using Euclidean distance:

$$\Pr ivacy - Level = \frac{1}{mn}\sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}\left|x_{ij} - y_{ij}\right|^2} \quad (3)$$

5.2 Global Perturbation Using *G* Algorithm

After executing the local obfuscation process, the global perturbation algorithm at the requester side is started. The idea is cluster multidimensional data using fast density clustering algorithm, then perturb each dimension in each cluster in such a way to preserve its range. In order to allow the global perturbation agent to execute *G* algorithm, we introduce an enhanced mean shift (EMS) algorithm which is tailored algorithm for the global perturbation phase that has advantage over previous algorithm proposed in [17] and it requires low computational complexity in clustering large data sets. we employ Gaussian *KD-tree* [18] clustering to reduce the feature space of the locally obfuscated data.

The G algorithm consists of two steps:

Step 1: Build different density based clusters

1. We build the tree in a top down manner starting from a root cell similar to [18, 19]. Each inner node of the tree S represents a *d*-dimensional cube cell which stores the dimension S_d along which it cuts, the cut value S_{cut} on that dimension, the bounds

of the node in that dimension S_{min} and S_{max}, and pointers to its children S_{left} and S_{right}. All points in the leaf nodes of the *kd* tree are then considered as a sample and the *kd-tree* stores *m* samples defined as $y_j^*, j = 1, \ldots, m$ that construct the reduced feature space of the original obfuscated data set.

2. Assign each record x_i to its nearest y_j based on *kd*-search, then compute a new sample, we called it $y_j^*, j = 1, \ldots, m$.
3. Generated y_j^* is a feature vector of *d*-dimensions, that is considered as a more accurate sample of the original obfuscated data set that will be used in the mean shift clustering.
4. The mean shift clustering iteratively performs these two steps:
   - Computation of mean shit vector based on the reduced feature space as following:

$$m(x_j) = \frac{\sum_{y_i^* \in N(y_j^*)} y_i^* g\left(\left\|\frac{x_j - y_i^*}{h}\right\|^2\right)}{\sum_{y_i^* \in N(y_j^*)} g\left(\left\|\frac{x_j - y_i^*}{h}\right\|^2\right)} - x_j, j = 1, 2..m \tag{4}$$

   Where $g(x) = -k'(x)$ defined when the derivate of function $k(x)$ exists, and $k(x), 0 \le x \le 1$ is called kernel function satisfying: $k(x) = c_{k,d} k\left(\|x\|^2\right) > 0$, $\|x\| \le 1$ and $\int_{-\infty}^{\infty} k(x) dx = 1$

   - Update the current position x_{j+1} as following:

$$m(x_{j+1}) = \frac{\sum_{y_i^* \in N(y_j^*)} y_i^* g\left(\left\|\frac{x_j - y_i^*}{h}\right\|^2\right)}{\sum_{y_i^* \in N(y_j^*)} g\left(\left\|\frac{x_j - y_i^*}{h}\right\|^2\right)}, j = 1, 2..m \tag{5}$$

   Until reaching the stationary point which is the candidate cluster centre. x_j will coverage to the mode in reduced feature space, finally we get approximate modes of original data defined as $z_x, x = 1, ..., k$.
5. Finally, the points which are in the mode are associated with the same cluster. Then we interpolate the computed modes in samples to the original obfuscated data by searching for the nearest mode z_x for each point x_i .

Step 2: Generating random points in each dimension range
For each cluster C , perform the following procedure.

1. Calculate the interquartile range for each dimension A_i .
2. For each element $e_{ij} \in A$, generate a uniform distributed random number r_{ij} in that range and replace e_{ij} with r_{ij} .

6 Experiments

The proposed algorithms are implemented in C++. We used message passing interface (MPI) for a distributed memory implementation of *G* algorithm to mimic a reliable distributed network of peers. We evaluated the proposed algorithms from two different aspects: privacy achieved and accuracy of results. The experiments presented here were conducted using the Movielens dataset [20]. The dataset contains users' ratings on movies using discrete values between 1 and 5. We follow the experiential scenarios presented in [14] We divide the data set into a training set and testing set. The training set is obfuscated then used as a database for the PCRS. Each rating record in the testing set is divided into a rated items t_u and unrated items r_u. The set $t_{u,i}$ is presented to the PCRS for making predication $p_{u,i}$ for the unrated items $r_{u,i}$ using the same algorithm in [21]. To evaluate the accuracy of generated predications, we used the mean average error (MAE) metric proposed in [22]. The first experiment performed on CBT algorithm to measure the impact of the varying portion size and number of core-points on privacy levels of the transformed ratings. To measure that we kept portion size constant with different number of core-points and then we vary portion size with constant number of core-points. Based on the results shown in figs (3) and (4), we can conclude that the privacy level increases when portion size is increasing. On the other hand, privacy level is reduced with increasing number of core-points as large number of rows used in *TLUT*. Each user in the network can control his privacy by diverging different parameters of *LLA* algorithm. Note that reducing the privacy level means less information loss in the collected ratings presented to PCRS. However this means the transformed ratings are similar to the original ratings, so the attacker can acquire more sensitive information.

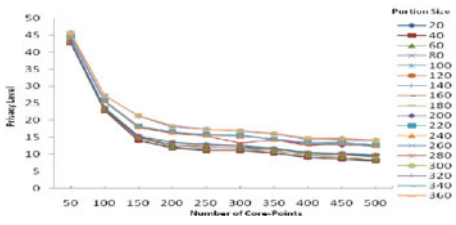


Fig. 3. Privacy level for different no.of core point

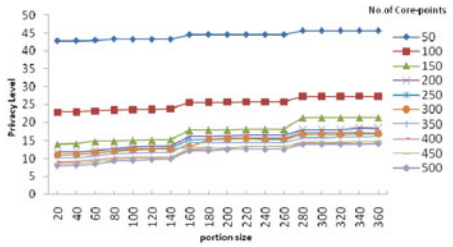


Fig. 4. Privacy level for different portion size

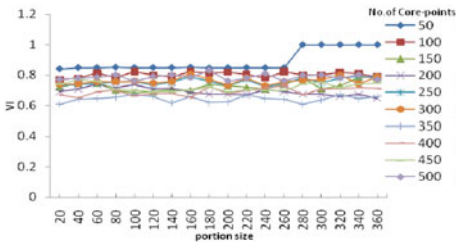


Fig. 5. VI for different portion size

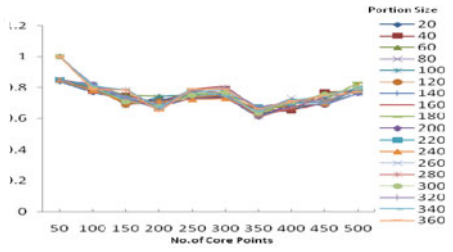


Fig. 6. VI for different no.of core points

To measure the distortion of the results, we use variation of information (VI) metric. Fig (5) shows VI for different number of core-points. One can see that at lower values of number of core-points the VI is high but slowly decreases with the increase in the number of core-points. At a certain point it rises to a local maxima then it decreases. Finally it rises again with the increasing number of core-points. We can justify that VI is high with fewer number of core-points as any point can move from one core-point to another. Moreover, with a plenty number of core-points there is a little chance of a point to move from one core-point to another, which causes increasing in VI values. The second experiment is performed on *G* algorithm to measure the impact of sample size on the accuracy level. We set a specific threshold value (100 users) for the minimum number of responding users for recommendation request. Otherwise the target user uses the PCRS obfuscated ratings database. As shown in fig (7) the increase in sample size leads to higher accuracy in the predications. However at a certain sample size, the accuracy of the predications starts decrement again due to the data loss in the sampling process.

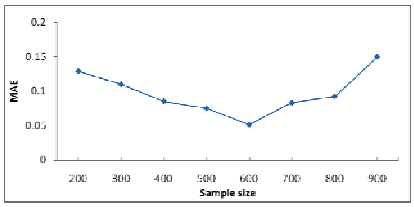


Fig. 7. Relation between sample size and MAE

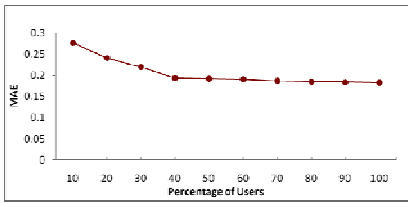


Fig. 8. Relation between Users and MAE

In the third experiment, we want to measure the impact of changing number of users involved in the group formation on the accuracy of the recommendations. We simulate a general case where the number of users was fixed to be 50.000. Then we assign different number of users to a certain recommendation request, and gradually increased the percentage of users who joined the request from 10% to 100% of them. We fixed the parameters for CBT and G algorithms then we measure MAE for the results. As shown in fig (8), the MAE value occurs at approximately 40% of the users are close to the MAE value for all users. Our conclusion is that, for low percentage of users the MAE value is close to the original MAE value for all users. As a result the target user does not need to broadcast the request to the full IPTV network to attain accurate results but he can employ multicast for certain users stored in his peer list to reduce the load in the network traffic. To illustrate the decrement of MAE values for recommendations based on diverse percentages of users groups and the whole users in the network, we calculated and plot fig (9). This verifies our conclusion that MAE approximately converges to the MAE which obtained using the whole users in our case. The final experiment was conducted to measure the impact of using CBT algorithm as a pre-processing step for G algorithm. As presented in fig (10), using CBT increases MAE values for lower percentages of participated users compared to using G algorithm alone. This can be explained, as the distortion effect of CBT algorithm will be clearly visible for a lower percentage of participating users.

However with the augment of percentage of users scale down the error in MAE values. So we can say that using the two stage obfuscation algorithms can increase the recommendation accuracy compared to only one stage based on one algorithm only.

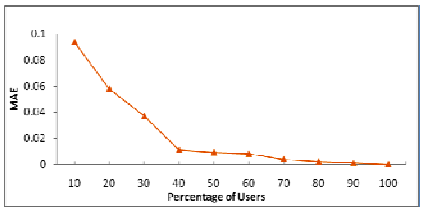


Fig. 9. The decrement of MAE values

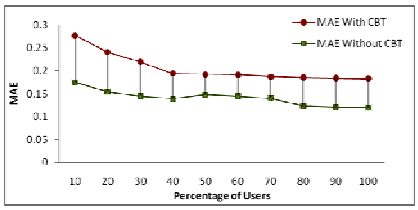


Fig. 10. The Influence of applying CBT algorithm

The results presented in these experiments show that the resulting dataset from our two stage obfuscation processes are quite similar in the accuracy of the generated recommendation to the original dataset. Our results also clarify that the proposed algorithms preserve the utility of the data to some degree such that to create reliable recommendations the target user does not have to collect profiles from numerous users, only a small percentage from users is need to attain that goal.

7 Conclusion and Future Wok

In this paper, we presented our ongoing work on building an agent based middleware to achieve privacy in recommender services. We gave an overview over the system components and recommendations process. Also we presented the novel algorithms that provide to users complete privacy over his profile privacy using two stage obfuscation processes. We test the performance of the proposed algorithms on real dataset and report the overall accuracy of the recommendations based on different privacy levels. The experiential results shows that preserving users' data privacy for in collaborative filtering recommendation system is possible and the mean average error can be reduced with proper tuning for algorithms parameters and large number of users. We need to perform extensive experiments in other real data set from UCI repository and compare the performance with other techniques, also we need to consider different data partitioning techniques, identify potential threats and add some protocols to ensure the privacy of the data against these threats.

References

1. Hand, S., Varan, D.: Interactive Narratives: Exploring the Links between Empathy, Interactivity and Structure, pp. 11–19 (2008)
2. Cranor, L.F.: 'I didn't buy it for myself' privacy and ecommerce personalization. In: Proceedings of the 2003 ACM Workshop on Privacy in the Electronic Society. ACM, Washington, DC (2003)
3. Dialogue, C.: Cyber Dialogue Survey Data Reveals Lost Revenue for Retailers Due to Widespread Consumer Privacy Concerns. Cyber Dialogue (2001)

4. Narayanan, A., Shmatikov, V.: Robust De-anonymization of Large Sparse Datasets. In: Proceedings of the 2008 IEEE Symposium on Security and Privacy. IEEE Computer Society (2008)
5. McSherry, F., Mironov, I.: Differentially private recommender systems: building privacy into the net. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 627–636. ACM, Paris (2009)
6. Esma, A.: Experimental Demonstration of a Hybrid Privacy-Preserving Recommender System. In: Gilles, B., Jose, M.F., Flavien Serge Mani, O., Zbigniew, R. (eds.), 161–170 (2008)
7. Canny, J.: Collaborative filtering with privacy via factor analysis. In: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 238–245. ACM, Tampere (2002)
8. Canny, J.: Collaborative Filtering with Privacy. In: Proceedings of the 2002 IEEE Symposium on Security and Privacy, p. 45. IEEE Computer Society (2002)
9. Polat, H., Du, W.: Privacy-Preserving Collaborative Filtering Using Randomized Perturbation Techniques. In: Proceedings of the Third IEEE International Conference on Data Mining, p. 625. IEEE Computer Society (2003)
10. Polat, H., Du, W.: SVD-based collaborative filtering with privacy. In: Proceedings of the 2005 ACM Symposium on Applied Computing, pp. 791–795. ACM, Santa Fe (2005)
11. Huang, Z., Du, W., Chen, B.: Deriving private information from randomized data. In: Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, pp. 37–48. ACM, Baltimore (2005)
12. Kargupta, H., Datta, S., Wang, Q., Sivakumar, K.: On the Privacy Preserving Properties of Random Data Perturbation Techniques. In: Proceedings of the Third IEEE International Conference on Data Mining, p. 99. IEEE Computer Society (2003)
13. Parameswaran, R., Blough, D.M.: Privacy preserving data obfuscation for inherently clustered data. Int. J. Inf. Comput. Secur. 2, 4–26 (2008)
14. Elmisery, A., Botvich, D.: Private Recommendation Service For IPTV System. In: 12th IFIP/IEEE International Symposium on Integrated Network Management. IEEE, Dublin (2011)
15. Blaze, M., Schneier, B.: The MacGuffin block cipher algorithm, pp. 97–110 (1995)
16. Elmisery, A.M., Huaiguo, F.: Privacy Preserving Distributed Learning Clustering Of HealthCare Data Using Cryptography Protocols. In: 34th IEEE Annual International Computer Software and Applications Workshops, Seoul, South Korea (2010)
17. Fukunaga, K., Hostetler, L.: The estimation of the gradient of a density function, with applications in pattern recognition. IEEE Transactions on Information Theory 21 (2003)
18. Xu, K., Li, Y., Ju, T., Hu, S.-M., Liu, T.-Q.: Efficient affinity-based edit propagation using K-D tree. In: ACM SIGGRAPH Asia 2009 Papers, pp. 1–6. ACM, Yokohama (2009)
19. Adams, A., Gelfand, N., Dolson, J., Levoy, M.: Gaussian KD-trees for fast high-dimensional filtering. ACM Trans. Graph. 28, 1–12 (2009)
20. Lam, S., Herlocker, J.: MovieLens Data Sets. Department of Computer Science and Engineering at the University of Minnesota (2006)
21. Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.: An algorithmic framework for performing collaborative filtering. In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 230–237. ACM, Berkeley (1999)
22. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. 22, 5–53 (2004)

Appendix B: IPTV Recommender Service Scenario

Article IV

Private Recommendation Service for IPTV Systems: Protecting User Profile Privacy

Ahmed M. Elmisery, Dmitri Botvich

In Proceedings of the 12th IFIP/IEEE International Symposium on Integrated Network Management (IM 2011), Dublin, Ireland, May 2011.

12th IFIP/IEEE IM 2011

Private Recommendation Service for IPTV Systems: Protecting User Profile Privacy

Ahmed M. Elmisery and Dmitri Botvich
Telecommunications Software & Systems Group
Waterford Institute of Technology, Waterford, Ireland
ahmedmohmed2001@gmail.com, dbotvich@tssg.org

***Abstract*—IPTV providers employ third party content recommendation service to help end users find personalized content, and at the same time increase content sales and gain competitive advantage over other IPTV providers. However current implementations of recommendation services are mostly centralized where all the information about the users' profiles is stored on a dedicated server. A common fear among users is that their user profile data being misused by recommendation provider. Also sharing profile data makes the end users vulnerable to attacks like insider attacks, where an employee of the recommendation service may compromise the confidentiality and integrity of their profiles. For these reasons, privacy aware users intentionally decline to use recommendation or even provide inaccurate or wrong information because they consider it as untrusted service. On the other hand, to build an accurate recommendation model the user must reveal information that is typically considered private such as watching history, previous buying behaviour, content ratings, etc. Further privacy concerns arise when the user data are stored in countries that have privacy laws different from the country where the service is consumed. This poses a severe privacy hazard, since the users profiles are fully under the control of recommendation provider and stored in locations that are not legally bound to ensure the privacy of its users. Due to different legal structures that relate to data privacy laws in different legal jurisdictions maintaining user profile privacy is not a trivial solution. Regardless of the official legal framework requirements, when outsourcing users' profiles the private data should be kept safe when it is in the possession of any third party service. In this paper we introduce a private recommendation method using collaborative filtering techniques. The method preserves the privacy of its users when using the system and allows sharing data among different users in the network. We also introduce two obfuscations algorithms that protect users profile privacy as well as preserve the aggregates in the dataset to maximize the utility of information to provide accurate recommendations. Using these algorithms provides the privacy of users personal profiles.**

Keywords-privacy; clustering; IPTV networks; recommender system

I. INTRODUCTION

Internet protocol television (IPTV) is one of the largest and rapidly growing services in the modern ICT. It distributes multimedia content (e.g. movies, news programs and documentaries) in digital format via broadband internet networks using packet switched network infrastructure [1]. Differently from conventional television, IPTV allows an interactive navigation of the available items [2]. Recently IPTV providers employ automated recommendation by collecting users' preferences for different content to create users' profile. The preferences of a user in the past can help the recommender service to predict other items that might be interested for him/her in the future. Collaborative filtering (CF) technique is utilized for recommendation purposes as one of the main categories of recommender systems. CF is based on the assumption that people with similar tastes prefer the same or similar items. In order to generate a recommendation, CF technique cluster users with the highest similarity in their interests. Then dynamic recommendations are served to them as a function of aggregate cluster interests. Thus there is a trade-off: the more users reveal information about their preferences, the more accurate recommendations are provided to them. However, at the same time the lower users' privacy levels can be guaranteed.

In real IPTV systems, it is natural that users might not be interested in providing their profiles to recommendation service or refrain from providing accurate information. This is mainly to their concern on privacy and the lack of laws that govern the use and distribution of such sort of data in some countries. Most service providers would try their best to keep the user's data private but occasionally, for example, when they are facing bankruptcy, they might sell the data to third parties in exchange of financial benefits. On the other hand, many services providers might violate users' privacy for their own commercial benefits. For instance, recently Amazon permitted the department of revenue to collect detailed information about purchase history of all North Carolina residents who purchased anything from Amazon since 2003 [3]. As a result, users concerned about their privacy refrain from providing the information to prevent potential exposure [4]. We can say that the privacy is the main concern for a significant number of users. In particular these concerns include user discomfort with the fact that a system gathers information about their preferences and purchase history at remote servers other than their own devices. The collected information breaches the privacy of the users on two levels.

1. The user's real identity is available to the remote service; that can associate the user's profile which contains his private information to his real identity. This is an obvious

978-1-4244-9220-6/11/$26.00 ©2011 IEEE

privacy breach, considering that a user does not want to reveal the link between his real identity and his profile. But users want to use the recommendation service.
2. If a user is not known to the service, it can try to deanonymize the user's identity by correlating the information contained in the user's profile and some information obtained from other databases [5] .

Recently, outsourcing services gain an attention with the advances of cloud computing as it enables SMEs that lack computational power or expertise to outsource their data to third party service providers to perform the required computation on the data. Privacy is the main concern for theses SMEs as service providers may be located abroad with different legislation and data privacy laws. The use of outsourcing in IPTV recommendation service is privacy hazard as both the data of IPTV provider (such as, product catalog) and user profiles are fully under the control of third party service provider. Even if there are legal guarantees from the third party provider to protect privacy, insider attacks are a new challenge that face this paradigm. Insider attacks are a privacy threat that needs to be addressed efficiently in order to provide advantage to IPTV recommendation service from outsourcing. There are a lot of recent cases of insider threats see e.g. [6, 7]. Recently p2p recommendation systems raised as a complete alternative to outsourced recommendation services, also it solve these privacy issues to a great extent [8], but it is not as accurate and practical as their centralized services Because it lack the complete information about all the users in the network.

In this paper we propose a framework for the recommendation service that bear in mind privacy related issues. It provides the utilization of users' profiles using a collaborative filtering technique that allows sharing profiles among different users in the network. We also present two stage obfuscations algorithms that protect user's privacy and preserve the aggregates in the dataset to maximize the utility of profiles to get accurate recommendations. The two stage obfuscation algorithms proposed to encounter the attack model presented in [9], where existing techniques has primarily considered a model where the attacker correlates obfuscated data with data from other publicly-accessible databases in order to reveal the sensitive information. In this work, we consider an attack model where the attacker colludes with some users in the network to obtain some partial information about the process used to obfuscate the data and/or some of the original data items themselves. The attacker can then use this partial information to attempt to reverse engineer the entire data set. Using our proposed algorithms, they give the user complete control on his personal profile. The user can make sure that the data does not leave his personal Set Top Box (STB) until it is properly desensitized. In the rest of this paper we will generically refer to news programs, movies and video on demand contents as items. In section II we describe some related work. In Section III we introduce an outsourced IPTV recommendation service scenario supporting our framework for the private recommender service (PRS). Section IV introduces the proposed obfuscation algorithms used in our framework. Section V describes some experiments and results based on obfuscation algorithms for the IPTV network. Section VI includes conclusions and future work.

II. Related Work

Majority of the existing Recommender systems are based on collaborative filtering, others focus on content based filtering using EPG data [10]. Users' profiles build upon ratings (explicit ratings procedures) or log archives (implicit ratings procedures) [11]. These procedures lead to two different approaches for the collaborative filtering including the rating based approaches and log based approaches. The majority of the literature addresses the problem of privacy on collaborative filtering technique, due to it is a potential source of leakage of private information shared by the users as shown in [12]. In [13] It is proposed a theoretical framework to preserve privacy of customers and the commercial interests of merchants. Their system is a hybrid recommender system that uses secure two party protocols and public key infrastructure to achieve the desired goals. In [14, 15] it is proposed a privacy preserving approach based on peer to peer techniques using users' communities, where the community will have a aggregate user profile representing the group as whole but not individual users. Personal information will be encrypted and the communication will be between individual users but not servers. Thus, the recommendations will be generated at client side. In [16, 17] it is suggested another method for privacy preserving on centralized recommender systems by adding uncertainty to the data by using a randomized perturbation technique while attempting to make sure that necessary statistical aggregates such as the mean don't get disturbed much. Hence, the server has no knowledge about true values of individual rating profiles for each user. They demonstrate that this method does not decrease essentially the obtained accuracy of the results. But recent research work [18, 19] pointed out that these techniques don't provide levels of privacy as it was previously thought. In [19] it is pointed out that arbitrary randomization is not safe because it is easy to breach the privacy protection it offers. They proposed a random matrix based spectral filtering techniques to recover the original data from perturbed data. Their experiments revealed that in many cases random perturbation techniques preserve very little privacy. Similar limitations were detailed in [18]. Storing user's profiles on their own side and running the recommender system in distributed manner without relying on any server is another approach proposed in [20], where authors proposed transmitting only similarity measures over the network and keep users profiles secret on their side to preserve privacy. Although this method eliminates the main source of threat against user's privacy, but it requires higher cooperation among users to generate useful recommendations. Secure recommendations using trust-based CRS have been proposed in [21], the approach provides protection from some types of attacks and achieves higher quality of the recommendations by using a web of trusted users whose ratings are preferred over untrusted users. Also it attempts to protect the personally identifiable fields of individuals participating in the ratings process. With a similar fashion to [15] the work in [22] considers encryption based approach for ratings. A recent work [23] proposed an efficient privacy preserving collaborative

recommender system based on the scalar product protocol using the Beaver's commodity model.

III. OUTSOURCED IPTV RECOMMENDATION SERVICE SCENIRO

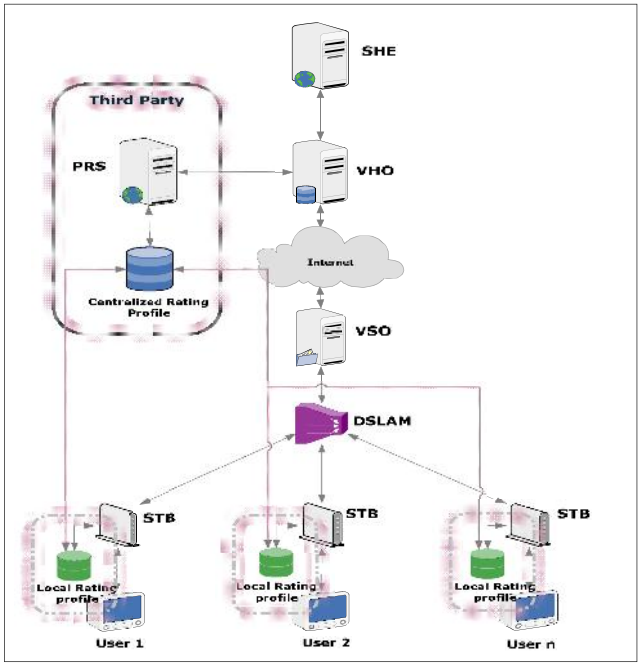


Figure 1: IPTV Network with Third Party Private Recommender Service

We consider the scenario where a private recommender service (PRS) is implemented on an external third party server and users give information about their items' ratings to the server in order to receive recommendations. We proposed a set of algorithms by which the users can modify their profiles to the privacy level they desire, and then submit the profile to the server. We don't assume the server to be completely honest. This is realistic assumption because the service provider needs to accomplish some business goals and increase its revenues. Intuitively, the system privacy is high if the PRS is not able to reconstruct the real ratings for users based on the information available to it. Figure (1) shows the architecture of our approach. Our Solution relies on hierarchical IPTV architecture. It consists of super head end (SHE) where all items are collected and distributed, Video Hub office (VHO) that receives content from SHE and distributes it to a number of video serving offices (VSO). VSO stores the content and distribute it to user's Set top box (STB). The Set top box is an electronic appliance that connects to both the network and the home television. With the advancement of data storage technology each STB is equipped with a mass storage, e.g. Cisco STB. In our framework, we will use the STB storage to store the local rating profile. On the other hand, video Hub office (VHO) maintains a centralized rating database that is used by the PRS. Additionally the entity operating the recommendation is a third-party recommender service provider connected to VHO that makes recommendations by consolidating the information received from multiple sources.

Figure 2: Recommenation Process with Two stage obfuscation

The obfuscated data are sent to PRS for making recommendations, which are then sent back to the corresponding user. We alleviate the user's identity problems stated above by using anonymous pseudonyms identities for users. The Recommendation Process shown in figure 2 can be summarized as follows:

1- The target user broadcasts a message to other users in the IPTV network to indicate his intention to start recommendations process. He also sanitizes his local profile using *BTA* algorithm.
2- Individual users that decide to respond to that request use the local obfuscation agent to obfuscate their local rating profiles based on *BTA* algorithm mentioned below. Then, they submit their locally obfuscated profiles to the requester.
3- If the size of group formation less than specific value, the target user contacts the PRS directly to gets recommendation from centralized rating profiles stored in it.
4- Otherwise, the target user incites his obfuscation agent to start global perturbation algorithm of the collected local rating profiles based on *EP* algorithm mentioned below.
5- In order to hide items that the group are interested in from the PRS, the target user masks the list of products rated by responding users using anonymous indexes which are linked to the actual items indexes through a secret map known only to the target user see table 1. One important issue is to standardize this secret map using hashing functions for group generated key or key generated by the VHO. As a result, the PRS will not be able to deal directly with items names but only with their hash values or anonymous indexes. Additionally the users' ratings are obfuscated from both PRS and VHO.

Anonymous index or Hash Value	Item index	Item Name	User1 rating	User2 rating	...	UserN rating
A_1	I_1	...	...	...		...
A_2	I_2	...	...	...		...
A_V	I_V	...	...	...		...

Table 1: Secret Map Used by Target User

6- The target user sends the globally perturbed ratings profiles together with pseudonyms of users participating in global perturbation process to PRS.
7- The PRS inserts pseudonyms into user database and their ratings into obfuscated rating database. Also, the PRS updates its model using received ratings and recalculates

core points for each cluster. The PRS computes the distance between the obfuscated user's profile and core points $Core-po\,\text{int}(C_{Xy})$ of each cluster for a certain $X \in \{1....N\}$, where N is the number of clusters and $y \in \{1.....j\}$, where y is the number of core-points in each cluster. The PRS uses clustering algorithm proposed in [24]. The PRS produces a list $\overline{A} = \{A_1, A_2, ..., A_i\}$ of anonymous indexes that users in cluster C_X have chosen in the past, as shown in table 2. Then PRS sends this list to the VHO that delivers it to the target user

User Cluster	Anonymous indexes
C_1	$A_1, A_8, A_6, A_{20}, ..., A_{13}$
C_2	$A_3, A_7, A_9, ... A_{80}$
C_N	$A_{11}, A_{10}, A_2, A_4, A_{33}, A_{27}, A_{21}, ..., A_{43}$

Table 2: Clustering Model in PRS

8- The target user incite his synchronize agent to submits the globally perturbed users' rating to the other users that participated with him in the process, so they able to update their rating data. Then the user unmasks the list $\overline{A}$ using his secret map to get finally list A. Finally the user selects the appropriate items for him from it.

The obfuscation agent in STB implements two stage obfuscation processes based on our proposed algorithms to achieve user privacy. These algorithms act as wrappers that obfuscate item's ratings before they are fed into the PRS. Since the databases are dynamic in nature, the obfuscation agent desensitizes the updated data periodically. Then the synchronize agent sends the updated data to the other users and PRS. So the recommendations are made on the most recent ratings. More information about PRS can be found in [25].

IV. Proposed Obfuscation Algorithms

In the next sections we present two different algorithms for the obfuscation agent to perturb the user profile in a way that secure user's ratings in the untrusted PRS with minimum loss of accuracy. In our framework, each user has two datasets representing his/her profile. Local rating profile: it represents the actual ratings of the user for different items, it is stored on STB. Each user perturbs his profile before merging it with similar users' profiles that rare willing to collaborate with him as part of the recommendation process. Centralized rating profile: this is the output of the two obfuscation processes where the user gets recommendation directly from the PRS that is based on previously collected ratings. We perform experiments on real datasets to illustrate the applicability of our algorithms and the privacy and accuracy levels achieved using them.

A. Local Obfuscation using blind tree algorithm (BTA)

We propose a novel pre-processing algorithm for profile obfuscation that has been designed especially for the sparse data problem we have here. We call this algorithm as the blind tree algorithm (BTA).We note that available anonymisation algorithms increase data distortion and, as result, will produce inaccurate recommendation model. We use local learning analysis (*LLA*) clustering method proposed in [24] to partition the dataset. After complete the partitioning, each point to be obfuscated is replaced by the average value of the nearest representative values in its cluster. If more core-points obtained by different user parameters, the profile will be partitioned into smaller clusters and the data distribution in the same cluster will be maintained. The output of our obfuscation algorithm should satisfy two requirements:

- Reconstructing the original profile from the obfuscated profile should be difficult, in order to preserve privacy.
- Preserve the distribution of the data to achieve accurate results for the recommendations.

Our algorithm consists of the following steps:

1. The users' ratings stored in STB as dataset D of c rows, where each row is a sequence of fields $X = x_1\ x_2\ x_3 \ldots\ldots x_m$.
2. Users' ratings dataset D is portioned into $D_1\ D_2\ D_3 \ldots\ldots D_n$ datasets of length L, if total number of attributes in the original dataset is not perfectly divisible by L then extra attributes are added with zero value which do not affect the result.
3. Generate the sets C_i and O_i for each D_i using *LLA* algorithm where C_i is the set of core-points and O_i is the rest of points in D_i. Note that *LLA* uses Gaussian Influence function as the similarity measure. Influence function between two data point x_i and x_j is given by
$$f_{Gauss}^{x_i}(x_j) = e^{-\frac{d(x_i,x_j)^2}{2\sigma^2}},$$
The field function for a candidate core-point is given by:
$$f_{Gauss}^{D}(x_j) = \sum_{s=1}^{k} e^{-\frac{d(x_j,x_{is})^2}{2\sigma^2}}$$
Clustering is performed on each D_i, resulting in k clusters $C_{i1}, C_{i2}, C_{i3}, ..., C_{ik}$ where each cluster is represented by its core-point, i.e. core-point of j^{th} cluster of i^{th} dataset is $(C_{ij}) = \{c_1, c_2, c_3, ...c_L\}$.So every point falls in exactly one cluster.
4. The initial tree T starts with a single parent node that represents all points in portion D_i. BTA starts the partitioning of D_i using points in C_i as splitting points. It computes the variance of each point in C_i based on points in current node, resulting in c_{ij*} which is the point with the maximum variance.
5. Find the median (interquartile range) of the cluster that have core-point (c_{ij*}), then partition the set of points in the current node into two subsets (child nodes) based on the median (interquartile range).
6. Repeat step 4 and 5 for all child nodes. The stop criterion for partitioning is when $C_i = \{\varnothing\}$.
7. Each point in O_i is obfuscated as following:

For each external node $o_{ij^*} = \{x'_{i1}, x'_{i2}, \ldots\ldots x'_{iv}\}$, $\forall x_{iv}$ is replaced with the averages $\overline{X} = 1/v \sum_{l=1}^{v} x'_{il}$ of all other points in the same external node. Repeating this step for each leaf node, the obfuscated portion of $D_i^{'}$ is formed from D_i. Then $D' = \bigcup_{i=1}^{n} D'_i$

8. Compute the privacy level by calculating the difference between the original dataset and obfuscated dataset using Euclidean distance:

$$\text{Privacy-Level} \frac{1}{m\ n} \sqrt{\sum_{i=1}^{c} \sum_{j=1}^{m} \left| 1_{ij} \right| x \Big|_{ij}^{2}}$$

B. Second:Global Perturbation using Enhanced Pertirbation (EP) Algorithm

After executing the local obfuscation process the global perturbation phase starts. The idea is to cluster the collected local rating profiles using *LLA* clustering algorithm and select the points with the highest field functions from each cluster to form a perturbation matrix then perform principle component analysis *(PCA)* [26] on that matrix. After that we perform the perturbation on each cluster in such a way to preserve its range. Finally we process the whole dataset with the perturbation matrix. The EP algorithm consists of the following steps:

1. We denote the collected users' profiles as dataset CD of c rows, where each row is sequence of fields $\mathrm{A} = \mathrm{A}_1\ \mathrm{A}_2\ \mathrm{A}_3 \ldots\ldots \mathrm{A}_m$.
2. The dataset CD is portioned using *LLA* algorithm into k clusters $CD_1, CD_2, CD_3, \ldots, CD_k$, where $CD_i = \{A_{i1}, A_{i2}, A_{i3}, \ldots, A_{io}\}$ and each cluster CD_i has j points with highest value for field function repressed by $H_i = \{C_{i1}, C_{i2}, C_{i3}, \ldots, C_{ij}\}$.
3. Form the perturbation matrix $PM = \{H_1, H_2, H_3, \ldots.. H_k\}$. If the sizes of sets H_i are not equal then *EP* chooses the next highest points from the clusters. Then we perform PCA on PM as following :
   - Compute the covariance matrix S of PM where:
   $$S_{ij} = 1/n-1 \sum_{p=1}^{n} \left(x_{pi} - \overline{x_i}\right)\left(x_{pj} - \overline{x_j}\right)$$
   - Compute the largest d eigenvalues $\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_d$ for S by $C\overline{\nu} = \lambda\overline{\nu}$.
   - Computer the eigenvectors of S_{ij} with d largest eignvalues. Then, the largest eigenvalue can be found using:
   $$w^* = \arg\max_{w:|w|=1} w^T C w,$$
   $$\lambda_{\max}(C) = \max_{w:|w|=1} w^T C w = w^{*T} C w^*$$
   The result is PM'.
4. *EP* enhances the approach proposed in [18] to perturb the points of each cluster as following:
   - First for each cluster, *EP* calculates the covariance matrix, and then decomposes covariance matrix to find the principle components in each cluster CD_k.
   - Project the points of each cluster into its principle components (that represents the direction where the data has the highest variance)
   - Calculate the interquartile range for CD_k, $\forall_{i=1}^{k} \forall_{j=1}^{o} a_{ij} \in CD_i$, then generate a uniform distributed random point r_{ij} in that range that substitute a_{ij} and project the resulting set back to the original coordinate system in order to obtain CD'_k.
5. Finally, project $CD' = \bigcup_{i=1}^{k} CD'_k$ on PM' to produce a lower dimensional dataset CD^L with L dimensions and The distribution of CD^L is similar to CD

V. EXPERIMENTS

The proposed algorithms are implemented in C++. We used message passing interface (MPI) for a distributed memory implementation of *EP* algorithm to mimic a distributed reliable network of peers. We evaluated the proposed algorithms from two different aspects: privacy achieved and accuracy of results. The experiments presented here were conducted using the Movielens dataset provided by Grouplens [27]. The dataset contains users' ratings on movies using discrete value between 1 and 5.

MovieLens Dataset	Users	Items	Ratings	Mean μ	Variance σ
	6040	3900	1000209	3.583	0.938

Table 3: Properties of experimental dataset

We start by analyzing the distribution of ratings among their values in the dataset; figure (3) gives these distributions. The experiments involve dividing the data set into a training set and testing set. The training set is obfuscated then used as a database for the PRS. Each rating record in the testing set is divided into rated items t_i and unrated items r_i. The set t is presented to the PRS for making predication p_i for the unrated items.

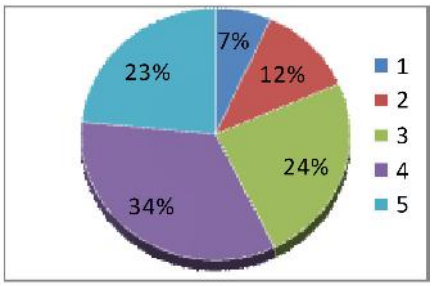


Figure 3: Distributions of ratings

To evaluate the accuracy of generated predications, we used the mean average error (MAE) metric proposed in [28]. The first experiment performed on *BTA* algorithm, we need to measure the impact of the varying portion size and number of splitting points on privacy levels of transformed dataset. In order to compute it, we keep portion size constant with different number of splitting points and then we vary portion size with constant number of splitting points. Based on the results shown in Figures (4) and (5); we can conclude that the privacy level increases when portion size is increasing. In the other hand the privacy level is reduced with increasing number of splitting points as fewer points in each node. This can be achieved by tuning different parameters in *LLA* algorithm. Note that the low privacy level reduces information loss when the PRS cluster different users but the transformed data is very similar to the original data. In this case an attacker can acquire more sensitive information. To measure the data error we use variation of information metric VI [29].

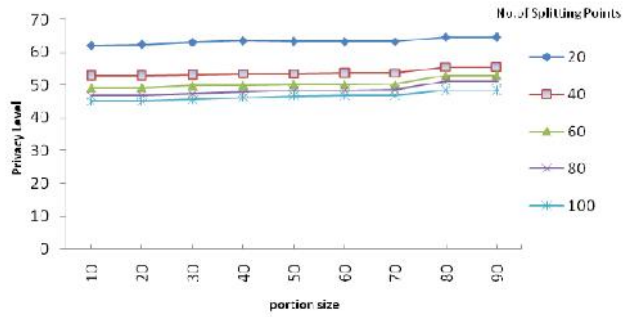


Figure 4: Privacy level for different partion size

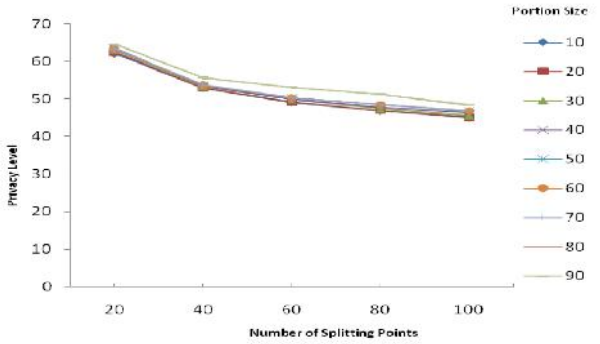


Figure 5: Privacy level for different no.of splitting points

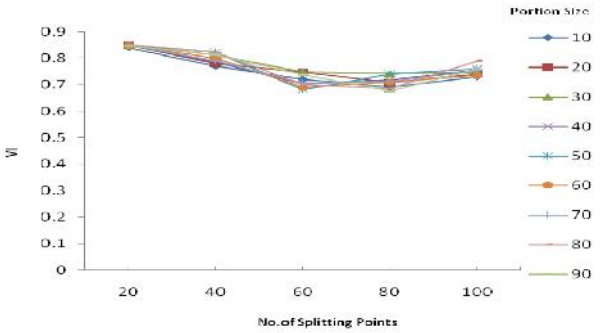


Figure 6: VI for different number of splitting points

Figure (6) shows VI for different values of splitting points. One can see that at low values of the number of splitting points VI is high but slowly decreases with increasing in the number of splitting points. At certain point it rises to local maxima then it decreases. Finally it rises again with the increasing of the number of splitting points. After the analysis of the results we can make the following conclusions:

- High VI at low values of the number of splitting points because there is a high chance of points to move from one node to another due to too low number of splitting points. As shown in Figure (4) privacy level is high at low values of the number of splitting points.
- High VI at high values of the number of splitting points is because less number of points in each node. So there is little chance of a point to move from one node to another due to high number of splitting points that might be more than optimal number of natural splitting points in the dataset.

In the second experiment performed on *EP* algorithm we measure the impact of *PM* size on accuracy level and the execution time for the global perturbation process. In order to compute both parameters we should have a specific threshold value that reflects minimum number of responding users to a target user to start the global perturbation process. We set it to 50 users, otherwise we use the PRS obfuscated ratings database. Figure (7) shows different execution time for *EP*-algorithm based on different values of the number of clusters. Our experiments with the *EP*-algorithm to measure the effect of *PM* size on the accuracy level is shown in Figure (8) illustrate that. We note that the increase in *PM* size leads to higher accuracy in the predications.

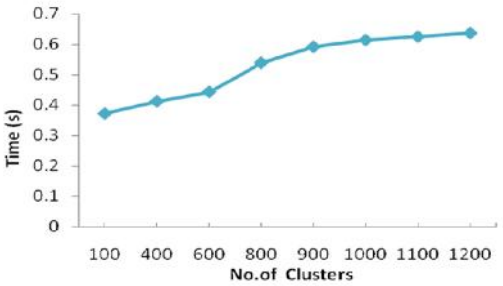


Figure 7: Execution time for EP

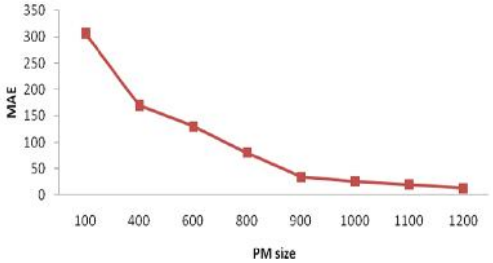


Figure 8: Relation between PM size and MAE

The final experiment performed to compute the impact of our two stage obfuscation algorithms on the accuracy of generated predications for different ratings groups. For this experiment, a set of 13702 ratings were separated from the MovieLens dataset. These ratings will be predicated using the PRS. We

repeat the predication experiment with increasing in the privacy level in *BTA* and *EP* algorithms then we compute MAE for these predictions. Figure (9) shows MAE values based on different privacy level.

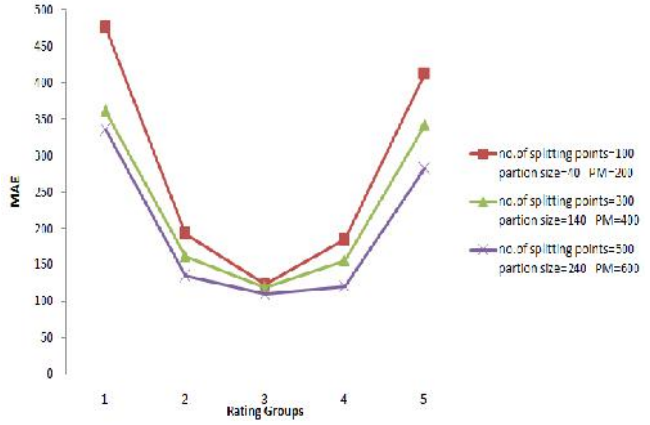


Figure 9: MAE for Different Ratings Groups

VI. CONCLUSION AND FUTURE WOK

In this paper, we presented our ongoing work on building a framework for private recommender service. We gave a brief overview of the recommendations process with application to IPTV. Also we presented the novel algorithms that provide to users complete privacy control over their profiles using two stage obfuscation processes. We test the performance of the proposed algorithms on real dataset. We evaluated how the overall accuracy of the recommendations depends on different privacy levels. The experiential results show that preserving users' data privacy for collaborative filtering recommendation system is possible. In particular mean average error can be reduced with proper tuning of the algorithms parameters for large number of users. We need to perform extensive experiments in other real data set from the UCI repository and compare the performance with other techniques. Also we need to consider different data partitioning techniques as well as identify potential threats and add some protocols to ensure the privacy of the data against those threats.

VII. ACKNOWLEDGMENT

This work has received support from the Higher Education Authority in Ireland under the PRTLI Cycle 4 programme, in the FutureComm project (Serving Society: Management of Future Communications Networks and Services). Special thanks to Mohamed Gaber from the School of Computing, University of Portsmouth, United Kingdom, for his valuable comments and feedback.

REFERENCES

[1] J. Jensen, "Interactive Television - A Brief Media History," ed, 2008, pp. 1-10.

[2] S. Hand and D. Varan, "Interactive Narratives: Exploring the Links between Empathy, Interactivity and Structure," ed, 2008, pp. 11-19.

[3] E. Schuman. (2010, *Amazon Bid to Make an (Expensive) Privacy Point*. Available: http://www.cbsnews.com/stories/2010/04/23/opinion/main6423734.shtml

[4] M. Hochhauser. (2000, *Privacy Rights Clearinghouse* Available: http://www.privacyrights.org/ar/amazon.htm

[5] A. Narayanan and V. Shmatikov, "Robust De-anonymization of Large Sparse Datasets," presented at the Proceedings of the 2008 IEEE Symposium on Security and Privacy, 2008.

[6] A. Hutton. (2010, 2010 Data Breach Investigations Report. *Research & Intelligence, Principal Verizon CybertrustSecurity*.

[7] R. Trzeciak. (2010, Risk Mitigation Strategies: Lessons Learned from Actual Insider Attacks. *CERT / Software Engineering Institute, Insider Threat Center*.

[8] B. Shlomo, *et al.*, "Privacy-Enhanced Collaborative Filtering," ed, 2005.

[9] R. Parameswaran and D. M. Blough, "Privacy preserving data obfuscation for inherently clustered data," *Int. J. Inf. Comput. Secur.*, vol. 2, pp. 4-26, 2008.

[10] L. Ardissono, *et al.*, *Personalized Digital Television: Targeting Programs to Individual Viewers (Human-Computer Interaction Series, 6)*: Kluwer Academic Publishers, 2004.

[11] M. d. Gemmis, *et al.*, "Preference Learning in Recommender Systems," presented at the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD), Slovenia, 2009.

[12] F. McSherry and I. Mironov, "Differentially private recommender systems: building privacy into the net," presented at the Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Paris, France, 2009.

[13] A. Esma, "Experimental Demonstration of a Hybrid Privacy-Preserving Recommender System," 2008, pp. 161-170.

[14] J. Canny, "Collaborative filtering with privacy via factor analysis," presented at the Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, Tampere, Finland, 2002.

[15] J. Canny, "Collaborative Filtering with Privacy," presented at the Proceedings of the 2002 IEEE Symposium on Security and Privacy, 2002.

[16] H. Polat and W. Du, "Privacy-Preserving Collaborative Filtering Using Randomized Perturbation Techniques," presented at the Proceedings of the Third IEEE International Conference on Data Mining, 2003.

[17] H. Polat and W. Du, "SVD-based collaborative filtering with privacy," presented at the Proceedings of the 2005 ACM symposium on Applied computing, Santa Fe, New Mexico, 2005.

[18] Z. Huang, *et al.*, "Deriving private information from randomized data," presented at the Proceedings of the 2005 ACM SIGMOD international conference on Management of data, Baltimore, Maryland, 2005.

[19] H. Kargupta, *et al.*, "On the Privacy Preserving Properties of Random Data Perturbation Techniques," presented at the Proceedings of the Third IEEE International Conference on Data Mining, 2003.

[20] B. N. Miller, *et al.*, "PocketLens: Toward a personal recommender system," *ACM Trans. Inf. Syst.*, vol. 22, pp. 437-476, 2004.

[21] P. Massa and P. Avesani, "Trust-aware recommender systems," presented at the Proceedings of the 2007 ACM conference on Recommender systems, Minneapolis, MN, USA, 2007.

[22] C.-L. A. Hsieh, *et al.*, "Preserving Privacy in Joining Recommender Systems," presented at the Proceedings of the 2008 International Conference on Information Security and Assurance (isa 2008), 2008.

[23] J. Zhan, *et al.*, "Privacy-Preserving Collaborative Recommender Systems," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on,* vol. 40, pp. 472-476, 2010.

[24] A. M. Elmisery and F. Huaiguo, "Privacy Preserving Distributed Learning Clustering Of HealthCare Data Using Cryptography Protocols," in *34th IEEE Annual International Computer Software and Applications*, Seoul, South Korea, 2010.

[25] A. M. Elmesiry and D. Botvich, "A Privacy Preserving Recommender System for IPTV," Waterford Institute of Technology Waterford, Technical Report 2010.

[26] A. Ben-Hur and I. Guyon, "Detecting Stable Clusters Using Principal Component Analysis," ed, 2003, pp. 159-182.

[27] S. Lam and J. Herlocker. MovieLens Data Sets [Online]. Available: http://www.grouplens.org/node/73

[28] J. L. Herlocker, *et al.*, "Evaluating collaborative filtering recommender systems," *ACM Trans. Inf. Syst.*, vol. 22, pp. 5-53, 2004.

[29] C. Kingsford, "Information Theory Notes," 2009.

Appendix B: IPTV Recommender Service Scenario

Article V

Privacy Aware Recommender Service for IPTV Networks

Ahmed M. Elmisery, Dmitri Botvich

In Proceedings of the 5th FTRA/IEEE International Conference on Multimedia and Ubiquitous Engineering (MUE 2011), Loutraki, Greece, June 2011.

2011 Fifth FTRA International Conference on Multimedia and Ubiquitous Engineering

Privacy Aware Recommender Service for IPTV Networks

Ahmed M. Elmisery and Dmitri Botvich
Telecommunications Software & Systems Group
Waterford Institute of Technology, Waterford, Ireland

***Abstract*—Providers of the next generation of IPTV services seek to gain competitive advantage over competing providers. In order to attract and satisfy customers, these providers must offer added value e.g. by delivering suitable content according to customers personal interests in a seamless way. This can achieved using recommender services. However, this brings about additional requirement related to the privacy of users' profiles that has to be addressed to make these services widely accepted. The ability to deploy a privacy aware recommender services and in the same time provide accurate recommendations become a key success for the spread of these services. Nevertheless, Current implementations of recommender services are mostly centralized where the users' profiles are stored in single server. This implementation fails to achieve the required privacy guarantee for the users, as it requires collecting accurate private information. In this paper, we present our efforts to build a private centralized recommender service (PRS) using collaborative filtering techniques by introducing an agent based middleware (AMPR) to ensure user profile privacy in the recommendation process. The driving force for using software agents in this work is the autonomic intelligent behaviour that can be achieved using agent technology. AMPR preserve the privacy of its users when using the system and allow private sharing of data among different user in the network. We also introduce two-stage obfuscation process embedded in AMPR that protect user profile privacy and preserve the aggregates in the dataset to maximize the usability of information in order to get accurate recommendations. These processes give the user a complete control on the privacy level of his personal profile. We also provide an IPTV network scenario and experimentation results**

Keywords-privacy; clustering; IPTV networks; recommender system; Multi-agent

I. INTRODUCTION

Providers of the next generation of IPTV services seek to gain competitive advantage over competing providers. In order to attract and satisfy customers, these services employ automated recommender systems to offer added value to their customers. Recommender service collects information about user preferences to create user profile. The preferences of a user in the past can help the recommender service to selecting the items that might be interested for him in the future.

While there are a number of solutions for recommendations services have been proposed and deployed for different domains like e-commerce and location based services, many of these systems have failed to meet the privacy requirements of service users, which result in a lack of acceptance of the respective services in general. A recent case in United Kingdom where the internet users report a series of privacy violation complaints to European commission about Phorm programme used at ISP level as it indulges into deep packet inspection of user traffic for targeted advertising purposes

Collaborative filtering (CF) techniques aim at countering information overload problem by extracting items that is relevant for a given user out of large body of items catalogue available via IPTV content provider. CF is based on the assumption that people with similar tastes prefer the same items. CF generates recommendations based on user profiles containing, for each user, personal data, preferences and rated items. The provided user profiles are usually structured and collected in two databases at the recommender service, namely user and ratings databases. CF techniques operate on these databases in order to generate recommendations of items that are probably relevant for a given user or determine users with the highest similarity in their interests.

Privacy is an essential concern in all recommender services as generating recommendations obviously requires the handling of user profiles. The more information is revealed to the recommender service about user profiles, the lower privacy levels can be guaranteed. According to surveys results [1], Privacy aware users refrain from providing accurate information because of their fears of personal safety and the lack of laws that govern the use and distribution of these data. In the other hand, many services providers might violate users' privacy for their own commercial benefits like Amazon. Based on a survey results in [2, 3] the users might leave a service provider because of privacy concerns. These concerns range from user discomfort with the fact that a system gathers information about his preferences and purchase history at remote servers other than his own device. However, privacy concerns should be balanced with other general requirements regarding performance and accuracy of recommendations as well.

In this paper we proposed an agent based middleware for private recommendation (AMPR) that bear in mind privacy issues related to the utilization of collaborative filtering technique in recommender services and allow private sharing of data among different users in the network. Our approach is based on software agent technology because fundamental features of agents such as autonomy, adaptability and the ability to communicate are essential requirements of our approach. We focus on stages related to users' profiles

978-0-7695-4470-0/11 $26.00 © 2011 IEEE
DOI 10.1109/MUE.2011.70

collection and processing and omit all aspects related to collaborative filtering, mainly because these stages are critical with regard to privacy as they involve different entities. We present two obfuscations algorithms that protect user privacy and preserve the aggregates in the dataset to maximize the usability of information in order to get accurate recommendations. Using these algorithms, give the user complete control on his personal profile, so he can make sure that the data does not leave his personal Set Top Box (STB) until it is properly desensitized. In the rest of this paper we will generically refer to news programs, movies and video on demand contents as Items. In section II describes some related work. In section III we introduce IPTV network scenario landing our private centralized recommender service (PRS). Section IV introduces the proposed obfuscation algorithms used in our framework. Section V describes some experiments and results based on the proposed obfuscation algorithms for IPTV network. Section VI includes conclusions and future work.

II. RELATED WORK

Majority of the existing Recommender systems are based on collaborative filtering, others focus on content based filtering using EPG data [4]. Collaborative filtering techniques build users' profiles in two ways upon ratings (explicit ratings procedures) or log archives (implicit ratings procedures) [5]. These procedures lead to two different approaches for the collaborative filtering including the rating based approaches and log based approaches. The majority of the literature addresses the problem of privacy on collaborative filtering technique, due to it is a potential source of leakage of private information shared by the users as shown in [6]. In [7] It is proposed a theoretical framework to preserve privacy of customers and the commercial interests of merchants. Their system is a hybrid recommender system that uses secure two party protocols and public key infrastructure to achieve the desired goals. In [8, 9] it is proposed a privacy preserving approach based on peer to peer techniques using users' communities, where the community will have a aggregate user profile representing the group as whole but not individual users. Personal information will be encrypted and the communication will be between individual users but not servers. Thus, the recommendations will be generated at client side. In [10, 11] it is suggested another method for privacy preserving on centralized recommender systems by adding uncertainty to the data by using a randomized perturbation technique while attempting to make sure that necessary statistical aggregates such as the mean don't get disturbed much. Hence, the server has no knowledge about true values of individual rating profiles for each user. They demonstrate that this method does not decrease essentially the obtained accuracy of the results. But recent research work [12, 13] pointed out that these techniques don't provide levels of privacy as it was previously thought. In [13] it is pointed out that arbitrary randomization is not safe because it is easy to breach the privacy protection it offers. They proposed a random matrix based spectral filtering techniques to recover the original data from perturbed data. Their experiments revealed that in many cases random perturbation techniques preserve very little privacy. Similar limitations were detailed in [12]. Storing user's profiles on their own side and running the recommender system in distributed manner without relying on any server is another approach proposed in [14], where authors proposed transmitting only similarity measures over the network and keep users profiles secret on their side to preserve privacy. Although this method eliminates the main source of threat against user's privacy, but it requires higher cooperation among users to generate useful recommendations. In this work, AMPR preserves the privacy of user profile form the attack model presented in [15]. The attack model for data obfuscation is different from the attack model for encryption-based techniques, but no common standard has been implemented for data obfuscation. Existing attack models has primarily considered a case where the attacker correlates obfuscated data with data from other publicly-accessible databases in order to reveal the sensitive information. But the attack model presented in [15], considers a case where the attacker colludes with some users in the network to obtain some partial information about the process used to obfuscate the data and/or some of the original data items themselves. The attacker can then use this partial information to attempt to reverse engineer the entire dataset.

III. PRIVATE RECOMMENDER SERVICE FOR IPTV NETWORK SCENIRO

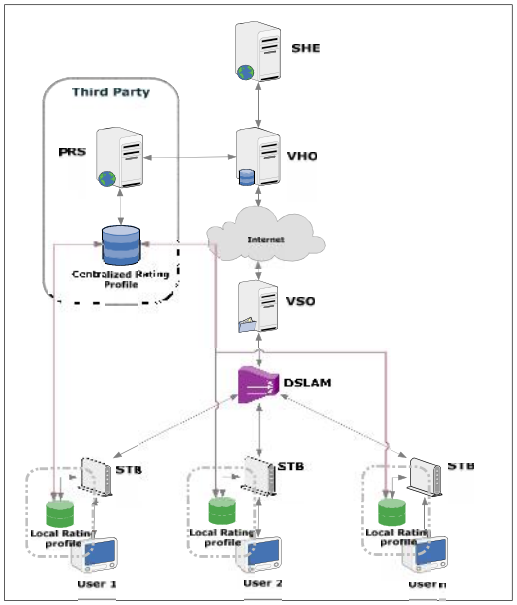


Figure 1: IPTV Network with Third Party Private Recommender Service

We consider the scenario proposed in [16, 17], where a private centralized recommender service (PRS) is implemented on an external third party server and users give information about their preferences to that server in order to receive recommendations. We proposed a set of algorithms by which the users can modify their profiles to the privacy

level they desire, and then submit the profile to the server. We don't assume the server to be completely malicious. This is a realistic assumption because the service provider needs to accomplish some business goals and increase its revenues. Intuitively, the system privacy is high if the PRS is not able to reconstruct the real ratings for users based on the information available to it. Figure (1) shows the architecture of our approach. Our Solution relies on hierarchical IPTV architecture. It consists of super head end (SHE) where all items are collected and distributed, Video Hub office (VHO) that receives content from SHE and distributes it to a number of video serving offices (VSO). VSO stores the content and distribute it to user's Set top box (STB). The Set top box is an electronic appliance that connects to both the network and the home television. With the advancement of data storage technology each STB is equipped with a mass storage, e.g. Cisco STB. In our framework, we will use the STB storage to store the user profile. On the other hand, video Hub office (VHO) maintains a centralized rating database that is used by the PRS. Additionally the entity operating the recommendation is a third-party recommender service provider connected to VHO that makes recommendations by consolidating the information received from multiple sources. We alleviate the user's identity problems stated above by using anonymous pseudonyms identities for users. The recommendation process based on the two stage obfuscation process in our framework can be summarized as following:

1. The target user broadcasts a message to other users in the IPTV network to indicate his intention to start recommendations process. He also used local obfuscation agent to sanitize his local profile.
2. Individual users that decide to respond to that request use their local obfuscation agent to obfuscate their local profiles based on *CTA* algorithm (mentioned below). Then, they submit their locally obfuscated profiles to the requester.
3. If the size of group formation less than specific value, the target user contacts the PRS directly to gets recommendation from the centralized profiles stored in it. Otherwise, the target user incites his global perturbation agent to start *EVS* algorithm (mentioned below) on the collected locally obfuscated profiles.
4. In order to hide items that the group are interested in from PRS, the target user masks the list of items rated by the responding users using anonymous indexes which are linked to the actual items indexes in the Video Hub office (VHO) through a secret map Ω know only by target user see table 1. One important issue to standardize this secret map is to use hashing functions using group generated key or key generated by the VHO to hash or mask his catalogue from the PRS.

Anonymous index or Hash Value	Item index	Item Name	User1 rating	User2 rating	...	UserN rating
A_1	I_1	...	...	...		...
A_2	I_2	...	...	...		...
A_v	I_r	...	...	...		...

Table 1: Secret Map Ω Used by Target User

Due to that the PRS will not be able to deal directly with items names but their hash values or anonymous index. Additionally the users' ratings are obfuscated from both PRS and VHO.

5. The target user sends the globally perturbed ratings profiles together with pseudonyms of users participate in global perturbation process to PRS.
6. The PRS insert pseudonyms into user database and their ratings into obfuscated rating database. PRS updates its model using received ratings, and then it produces a list $\overline{A_i} = \{A_1, A_2, A_3 \ldots . A_y\}$ of anonymous indexes that users in the same cluster have chosen in the past. The PRS submit that list to target user.
7. The target user then submits the globally perturbed profiles to the other users that participate with him in the process, so they able to update their rating data. Then he unmasks the list $\overline{A_i}$ using his secret map Ω to get finally list A_i , then he selects the appropriate items for him from it.

IV. Proposed Obfuscation Algorithms

In the next subsections, we present two different algorithms used by the obfuscation agents in AMPR to obfuscate the user profile in a way that secure user's ratings in the untrusted PRS with minimum loss of accuracy. In our framework, each user has two datasets representing his/her profile. Local profile: it represents the actual ratings of the user for different items; it is stored on his STB. Each user obfuscates his local profile before merging it with similar users' profiles that rare willing to collaborate with him as part of the recommendation process. A centralized profile: this is the output of the two-stage obfuscation process where the user gets recommendation directly from the PRS based on the previously collected profiles. We perform experiments on real datasets to illustrate the applicability of our algorithms and the privacy and accuracy levels achieved using them.

A. Local Obfuscation using Clustering Transformation Algorithm (CTA)

We propose a novel algorithm for obfuscating the user profile before sharing it with other users in the IPTV network. This algorithm called *CTA*, which has been designed especially for the sparse data problem we have here. We noted that, the available anonymisation algorithms increase data distortion and, as result inaccurate recommendation model could constructed. Maintaining utility and privacy for profiles seems to be contradictory goals to attain. *CTA* partitions the user profile into smaller clusters and then pre-process each cluster such that the distances inside the same cluster will maintained in its obfuscated version. We use local learning analysis (*LLA*) clustering method proposed in [18] to partition the dataset. After complete the partitioning, we embed each cluster into a random dimension space so the sensitive ratings will be protected. Then the resulting cluster will be rotated randomly. In such a way, *CTA* obfuscates the data inside user profile while preserving the distances between the data

points to provide high accurate results when performing recommendations. The output of our obfuscation algorithm should satisfy two requirements:

- Reconstructing the original profile from the obfuscated profile should be difficult, in order to preserve privacy.
- Preserve the distances of the data to achieve accurate results for the recommendations.

Our algorithm consists of the following steps:

1. The user ratings is stored in STB as dataset D consists of c rows, where each row is a sequence of X attributes where $X = x_1\ x_2\ x_3 \dots \dots \dots\ x_n$.
2. The dataset D is portioned vertically into $D_1\ D_2\ D_2 \dots \dots D_m$ subsets of length L, if n/L is not perfectly divisible then *CTA* randomly selects attributes already assigned to any subset and joins them to the attributes of the incomplete subsets.
3. Cluster each subset $\forall_{j=1}^{m} D_j$ Using *LLA* algorithm, which resulting in K clusters $D_j = C_{j1}, C_{j2}, C_{j3} \dots . C_{jk}$ for each subset. Note that *LLA* uses Gaussian Influence function as the similarity measure. Influence function between two data point x_i and x_j is given by
$$f_{Gauss}^{x_i}(x_j) = e^{-\frac{d(x_i,x_j)^2}{2\sigma^2}}$$
And the field function for a candidate core-point selection, which is given by:
$$f_{Gauss}^{D}(x_j) = \sum_{s=1}^{k} e^{-\frac{d(x_j,x_{is})^2}{2\sigma^2}}$$
So every point in the original dataset D falls exactly in one cluster. The aim of this step is to increase the privacy level of the transformation process and make reconstruction attacks difficult.
4. *CTA* generates two sets for each cluster $\forall_{i=1}^{k} C_{ji}$ in the subset D_j these are H_{ji} and O_{ji}. Where H_{ji} is the set of points with highest values for field function and O_{ji} is the rest of points in C_{ji}. For each point $x_{1i} \in H_{ji}$ construct a weighted graph Γ_i that contains its k-nearest neighbours in O_{ji}, each edge $e \in \Gamma_i$ has a weight equals to $f_{Gauss}^{b_{1i}}(x_{1i})$.
5. Estimate the geodesic distances by Computing the shortest distance between each two points in graph Γ_i using Dijkstra or Floyd algorithm and then build a distance matrix $D_{\Gamma_i} = \{ f_{Gauss}^{b_i}(x_i) \}$.
6. Based on D_{Γ_i}, we find a d-dim embedding space C'_{ji} using classical *MDS* [19] as follows
   - Calculate the matrix of squared distances $S = D_{\Gamma_i}^2$ and the centering matrix $H = 1 - 1/N\ ee^T$
   - The characteristic vectors are chosen to minimize $E = \left\| \tau(D_{\Gamma_i}) - \tau(D_d) \right\|_{L^2}$, where $\tau(D_d)$ is the distance matrix for the *d-dim* embedding space, and τ converts distances to inner products $\tau = -HSH/2$.
7. For each cluster $\forall_{j=1}^{m} \forall_{i=1}^{k} C'_{ji}$, *CTA* randomly select two attributes x_a and x_b to perform rotation perturbation on selected attributes $R(x_a, x_b)$ using transformation matrix M_j^{θ} setup by the user for each cluster using range of angles defined in advance by the user.
8. Repeating steps 4-7 for all clusters in $\forall_{j=1}^{m} D_j$ to get the obfuscated portion D'_j. Finally, the obfuscated dataset is obtained by $D' = \cup_{j=1}^{n} D'_j$.

B. *Global Perturbation using Enhanced Value-Substitution (EVS) Algorithm*

After executing the local obfuscation process the global perturbation phase starts. The key idea for *EVS* is based on the work in [20] that uses Hilbert curve as a dimensionality reduction tool to create a cloaking regions to attain privacy for users. Hilbert curve also has the ability to maintain the association between different dimensions. In this subsection, we extend this idea as following, we also use Hilbert curve to map m-dimensional profile to 1-dimensional profile then *EVS* discovers the distribution of that1-dimensional profile. Finally, we perform perturbation based on that distribution in such a way to preserve the profile range. The steps for *EVS* algorithm consists of the following steps:

1. We denote the collected m-dimensional user profiles as dataset D of c rows, where each row is a sequence of m dimensions $A = A_1, A_2, A_3, A_4 \dots \dots, A_m$.
2. *EVS* divides the m-dimensional profile into grids of order k (where k is user defined value) as shown in [20, 21]. For order k, the range for each dimension divided into 2^k intervals.
3. For each dimension $\forall_{i=1}^{m} A_i$ of the collected profile D:
   - Compute the *k*-order Hilbert value for each data point $\forall_{x=1}^{c} a_{ix}$. This value represents the index of the corresponding interval where it falls in.
   - *EVS* sort the Hilbert values from smallest to biggest, then use the step length (a user defined parameter) to measure whether any two values are near from each other or not. If these values are near, they are placed in the same partition $\forall_{v=1}^{k} k_{iv}$.

   These two steps iterates for all m-dimensions. The final result from these steps is k partitions for each dimension denoted as $\forall_{i=1}^{m} \forall_{v=1}^{k} C_{iv}$
4. *EVS* constructs a N shared nearest neighbour sets S_r where $r = 1 \dots . N$ as in [22] from different partitions with a new modified similarity function as following, two partitions in different dimensions C_{iv}, C_{i+1v} form a shared nearest neighbour set S_r if they share k-number of common elements such that $S_r = C_{iv} \cup C_{i+1v}$
5. For each newly created set S_r, *EVS* calculates its interquartile range. Then, for each point $a_i \in S_r$ generate a uniform distributed random point n in that range that can substitutes a_i.
6. Finally, the new set $D' = \cup_{r=1}^{N} S_r$ is sent to PRS

V. EXPERIMENTS

The proposed algorithms are implemented in C++. We used message passing interface (MPI) for a distributed memory implementation of *EVS* algorithm to mimic a distributed reliable network of peers. We evaluated the

proposed algorithms from two different aspects: privacy achieved and accuracy of results. The experiments presented here were conducted using the Movielens dataset provided by Grouplens [23]. The dataset contains users' ratings on movies using discrete value between 1 and 5. The data in our experiments consists of 100.000 ratings for 1.682 items by 943 users. The experiments involve dividing the data set into a training set and testing set. The training set is obfuscated then used as a database for the PRS. Each rating record in the testing set is divided into rated items t_i and unrated items r_i .The set t is presented to the PRS for making predication $p_i$for the unrated items. To evaluate the accuracy of generated predications, we used the mean absolute error (MAE) metric proposed in [24]. MAE is one of most famous metrics for recommendation quality. We can define it as following: Given a user predicated ratings set $p = \{p_1, p_2, p_3 \dots p_N\}$ and the corresponding real ratings set $r = \{r_1, r_2, r_3 \dots . r_N\}$, MAE is:

$$MAE = \sum_{i=1}^{N} |p_i - r_i| / N$$

MAE measures the predication verity between the predicated ratings and the real ratings, so smaller MAE means better recommendations provided by PRS. To measure the privacy or distortion level achieved using our algorithms, we use variation of information metric VI [25] to estimate data error. VI is:

$$VI = H(p) + H(r) - 2I(p, r)$$

Here $H(p)$ is entropy of p, r and $I(p, r)$ is mutual information between p and r .The higher VI means the larger distortion between the obfuscated and original dataset, which means higher privacy level.

To evaluate the accuracy of *CTA* algorithm with respect to different number of dimensions in user profile, we control *d-dim* parameters of *CTA* to vary number of dimensions during the evaluation. Figure (1) shows the performance of recommendations of locally obfuscated data, as shown the accuracy of recommendations based on obfuscated data is little bit low when the dimension is low. But at a certain number of dimensions (500), the accuracy of recommendations of obfuscated data is nearly equal to the accuracy obtained using original data.

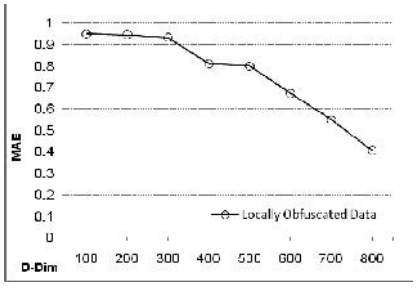


Figure 2: Accuracy of recommendations for obfucated dataset using *CTA*

In the second experiment performed on *CTA* algorithm, we examine the effect of *d-dim* on VI values. As shown in figure (3), VI values decrease with respect to the increase in *d-dim* values in user profile. *d-dim* is the key element for privacy level where smaller *d-dim* value, the higher VI values (privacy level) of *CTA*. However, clearly the highest privacy is at *d-dim*=100. There is a noticeable drop of VI values when we change *d-dim* from 300 to 600.*d-dim* value 400 is considered as a critical point for the privacy.Note that rotation transformation adds extra privacy layer to the data and in the same time maintains the distance between data points to enable PRS to build accurate recommendation models.

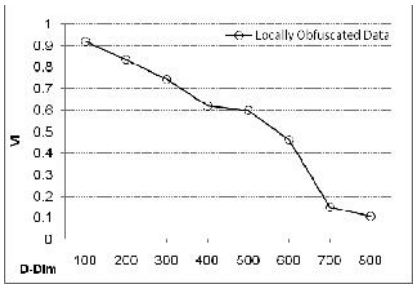


Figure 3: Privacy levels for the obfucated dataset using *CTA*

In the first experiment performed on *EVS* algorithm, we measured the relation between different Hilbert curve parameters (order and step length) on the accuracy and privacy levels attained. We map the locally obfuscated dataset to Hilbert values using order 3, 6 and 9. We gradually increased the step length from 10 to 80. Figure (4) shows the accuracy of recommendations based on different step length and curve order. We can see that as the order increases, the obfuscated data can offer better predictions for the ratings. This is because as the order has higher value, the granularity of the Hilbert curve becomes finer. So, the mapped values can preserve the data distribution of the original dataset. On the other hand, selecting larger step length increases MAE values as large partitions are formed with higher range to generate random values from it, such that these random values substitute real values in the dataset.

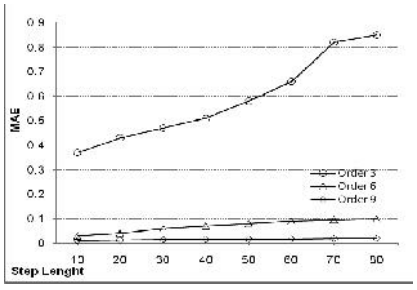


Figure 4: Accuracy level for different step length and orders for *EVS*

As for the privacy as shown in figure (5), when the order increases a smaller range is calculated within each partition which introduces less substituted values compared with lower orders that attain higher VI values. The reason for this is larger order divides the m-dimensional profile into more grids, which makes Hilbert curve better reflects the data distribution. Also, we can see that for the same Hilbert curve order the VI values are generally the same for different step length except for order 3, in which VI values has a sharp

increase when step length grows from 50 to 60.The effect of increasing step length on VI values is more sensible in lower curve orders as fewer girds are formed and the increase of step length covers more portions of them, which will introduce a higher range to generate random values from it. So the target user should select *EVS* parameters in such a way to achieve a trade off between privacy and accuracy.

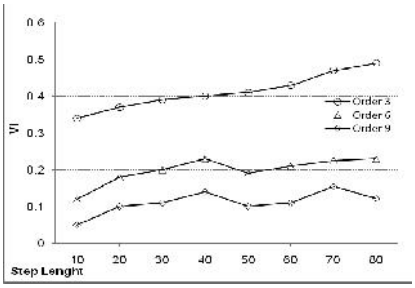


Figure 5: Privacy level for different step length and orders for *EVS*

We continued our experiments with *EVS* algorithm; we measured the execution time for *EVS* as it is executed locally at the target user's STB box on the collected datasets. The execution time for *EVS* is composed of the time to get partitions based on Hilbert curve and the time to generate random noise. The results for the execution time are shown in figure (6). We can see that as the order of Hilbert curve goes higher, the execution time generally increased than that for a lower order. This growth because of the time consumed in mapping data points to different Hilbert values is dependent on curve order. For different step lengths, the executions time various without substantial trend. As the step length only determines the size of partitions in each dimension; finding these partitions are only dependant on the number of dimensions.

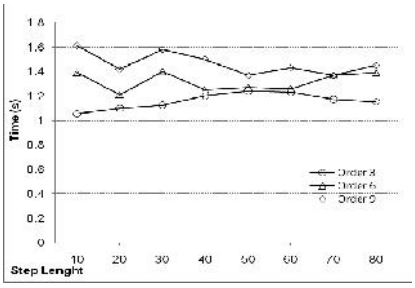


Figure 6: Execution time for different step length in *EVS*

Finally, we measured the overall recommendation accuracy of our two stage obfuscation process on the same dataset. Figure (7) shows the accuracy attained using one stage obfuscation (*EVS* algorithm) and two stage obfuscation process (*CTA+EVS* algorithms) for the same dataset. For *EVS* algorithm in both experiments, we set the curve order to be 2 and gradually increases step length respectively. We obfuscate the original dataset using *EVS* algorithm then fed it to *PRS*. In the same manner, we repeated that but obfuscate the dataset using both *CTA* and *EVS* algorithms. We calculated MAE for the resulting recommendations of both experiments; we can see the increase of MAE values on the obfuscated dataset due to two stage obfuscation process. On the other hand, Increasing step length, attain high MAE values for original data but has marginal impact on obfuscated data, this growth due to the formation of large partitions with outliers that influence on the calculation of interquartile range. That case is not applied on obfuscated data, as they already pre-processed using *CTA*.

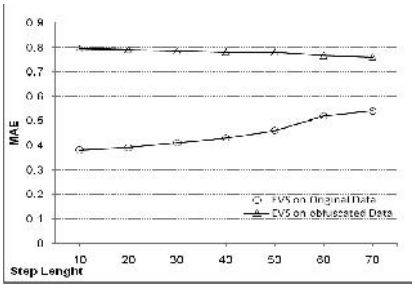


Figure 7: MAE comparsion between *EVS* on the original and obfucated datasets

VI. CONCLUSION AND FUTURE WOK

In this paper, we presented our ongoing work on building an agent based middleware for private recommendations service. We gave a brief overview of the recommendations process with application to IPTV. Also we presented the novel algorithms that provide to users complete privacy control over their profiles using two stage obfuscation process. We test the performance of the proposed algorithms on real dataset. We evaluated how the overall accuracy of the recommendations depends on different privacy levels. The experiential results show that preserving users' data privacy for collaborative filtering recommendation system is possible. In particular mean average error can be reduced with proper tuning of the algorithms parameters for large number of users. We realized that there are many challenges in building a private recommender service. As a result we focused in an agent based middleware scenario. This allow us to move forward in building an integrated system while studying issues such as a dynamic data release at a later stage and deferring certain issues such as virtualized schema and auditing to future research agenda. We need to perform extensive experiments in other real data set from the UCI repository and compare the performance with other techniques. Also we need to consider different data partitioning techniques as well as identify potential threats and add some protocols to ensure the privacy of the data against those threats.

VII. ACKNOWLEDGMENT

This work has received support from the Higher Education Authority in Ireland under the PRTLI Cycle 4 programme, in the FutureComm project (Serving Society: Management of Future Communications Networks and Services).

REFERENCES

[1] M. Teltzrow and A. Kobsa, "Impacts of user privacy preferences on personalized systems: a comparative study," in *Designing personalized user experiences in eCommerce*, ed: Kluwer Academic Publishers, 2004, pp. 315-332.

[2] L. F. Cranor, "'I didn't buy it for myself' privacy and ecommerce personalization," presented at the Proceedings of the 2003 ACM workshop on Privacy in the electronic society, Washington, DC, 2003.

[3] C. Dialogue, "Cyber Dialogue Survey Data Reveals Lost Revenue for Retailers Due to Widespread Consumer Privacy Concerns," in *Cyber Dialogue*, ed, 2001.

[4] L. Ardissono, *et al.*, *Personalized Digital Television: Targeting Programs to Individual Viewers (Human-Computer Interaction Series, 6)*: Kluwer Academic Publishers, 2004.

[5] M. d. Gemmis, *et al.*, "Preference Learning in Recommender Systems," presented at the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD), Slovenia, 2009.

[6] F. McSherry and I. Mironov, "Differentially private recommender systems: building privacy into the net," presented at the Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Paris, France, 2009.

[7] A. Esma, "Experimental Demonstration of a Hybrid Privacy-Preserving Recommender System," 2008, pp. 161-170.

[8] J. Canny, "Collaborative filtering with privacy via factor analysis," presented at the Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, Tampere, Finland, 2002.

[9] J. Canny, "Collaborative Filtering with Privacy," presented at the Proceedings of the 2002 IEEE Symposium on Security and Privacy, 2002.

[10] H. Polat and W. Du, "Privacy-Preserving Collaborative Filtering Using Randomized Perturbation Techniques," presented at the Proceedings of the Third IEEE International Conference on Data Mining, 2003.

[11] H. Polat and W. Du, "SVD-based collaborative filtering with privacy," presented at the Proceedings of the 2005 ACM symposium on Applied computing, Santa Fe, New Mexico, 2005.

[12] Z. Huang, *et al.*, "Deriving private information from randomized data," presented at the Proceedings of the 2005 ACM SIGMOD international conference on Management of data, Baltimore, Maryland, 2005.

[13] H. Kargupta, *et al.*, "On the Privacy Preserving Properties of Random Data Perturbation Techniques," presented at the Proceedings of the Third IEEE International Conference on Data Mining, 2003.

[14] B. N. Miller, *et al.*, "PocketLens: Toward a personal recommender system," *ACM Trans. Inf. Syst.,* vol. 22, pp. 437-476, 2004.

[15] R. Parameswaran and D. M. Blough, "Privacy preserving data obfuscation for inherently clustered data," *Int. J. Inf. Comput. Secur.,* vol. 2, pp. 4-26, 2008.

[16] A. Elmisery and D. Botvich, "Private Recommendation Service For IPTV System," in *12th IFIP/IEEE International Symposium on Integrated Network Management*, Dublin, Ireland, 2011.

[17] A. Elmisery and D. Botvich, "Agent Based Middleware for Maintaining User Privacy in IPTV Recommender Services," in *3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems*, Aalborg, Denmark, 2011.

[18] A. Elmisery and F. Huaiguo, "Privacy Preserving Distributed Learning Clustering Of HealthCare Data Using Cryptography Protocols," in *34th IEEE Annual International Computer Software and Applications Workshops*, Seoul, South Korea, 2010.

[19] I. Borg and P. J. F. Groenen, *Modern Multidimensional Scaling: Theory and Applications (Springer Series in Statistics)*: Springer, 2005.

[20] G. Ghinita, *et al.*, "PRIVE: anonymous location-based queries in distributed mobile systems," presented at the Proceedings of the 16th international conference on World Wide Web, Banff, Alberta, Canada, 2007.

[21] A. Reaz and B. Raouf, "A Scalable Peer-to-peer Protocol Enabling Efficient and Flexible Search," ed, 2010.

[22] R. A. Jarvis and E. A. Patrick, "Clustering Using a Similarity Measure Based on Shared Near Neighbors," *IEEE Trans. Comput.,* vol. 22, pp. 1025-1034, 1973.

[23] S. Lam and J. Herlocker. MovieLens Data Sets [Online]. Available: http://www.grouplens.org/node/73

[24] J. L. Herlocker, *et al.*, "Evaluating collaborative filtering recommender systems," *ACM Trans. Inf. Syst.,* vol. 22, pp. 5-53, 2004.

[25] C. Kingsford, "Information Theory Notes," 2009.

Appendix B: IPTV Recommender Service Scenario

Article VI

Enhanced Middleware for Collaborative Privacy in IPTV Recommender Services

Ahmed M. Elmisery, Dmitri Botvich

In the KITCS/FTRA International Journal of Convergence, Volume 2, Number 2, December 2011.

Volume 2, Number 2, December 2011 *Journal of Convergence*

Enhanced Middleware for Collaborative Privacy in IPTV Recommender Services

Ahmed M. Elmisery* and Dmitri Botvich
Telecommunications Software & Systems Group-TSSG
Waterford Institute of Technology-WIT, Co. Waterford, Ireland
*ahmedmohmed2001@gmail.com

***Abstract*—One of the concerns users have to confront when using an IPTV system is information overload, which makes it difficult for them to find suitable content according to their personal preferences. A recommendation service is one of the most widely adopted technologies for alleviating this problem; these services intend to provide people with referrals of items that they will appreciate based upon their preferences. IPTV users must ensure that their sensitive preferences collected by any recommendation service are properly secured. In this work, we introduce a framework for a private recommender service based on Enhanced Middleware for Collaborative Privacy (*EMCP*). *EMCP* executes a two-stage concealment process that gives the user complete control over the privacy level of his profile. We utilise a trust mechanism to augment the recommendation's accuracy and privacy. These trust heuristic spot users whom are trustworthy with respect to the user requesting the recommendation (target-user). Later, the neighbourhood formation is calculated using proximity metrics based on these trustworthy users. Finally, users submit their profiles in an obfuscated form without revealing any information about their data, and the computation of recommendations proceeds on the obfuscated data using a secure multi-party computation protocol. We expand the obfuscation scope from a single obfuscation level for all users to arbitrary obfuscation levels based on different trust levels between users. In other words, we correlate the obfuscation level with different levels of trust, so the more trusted a target user is, the less obfuscatedcopy of users' profiles he can access. We also provide an IPTV network scenario and experimentation results. Our results and analysis show that our two-stage concealment process not only protects the users' privacy, but can also maintain the accuracy of the recommendations.**

Keywords- Privacy; Clustering; IPTV Network; Recommendation-Services; Multi-agent

I. INTRODUCTION

Internet Protocol Television (IPTV) is a video service providing IP broadcasts and video on demand (VOD) over a broadband IP content delivery network (CDN) specialized in video services. The IPTV user has access to myriads of video content spanning IP broadcasts and VOD [1]. In this context it is difficult for them to find content that matches their preferences from the huge amount of video content available. In order to attract and satisfy these users, IPTV service providers employ recommendation services to increase their revenues and offer added value to their patrons. A recommendation service is a promising personalised service for IPTV users which offer referrals to users by building up users' profiles (explicit or implicit) based on their past ratings, behaviour, purchase history or demographic information. In the context of this work, a profile is a list that comprises the video content the user has watched or purchased, combined with their meta-data extracted from the content provider (i.e. genres, directors, actors and so on) and the ratings the user gave to such content.

Recommendation services are usually served using collaborative filtering (CF) algorithms, which is a popularly used technical approach to automate the word-of-mouth process; it is based on the hypothesis that people with similar tastes prefer the same items. The recommendations using the CF technique involves a server or main entity that collects users' profiles to find users that are similar to the user receiving the recommendation (target user), and then it executes CF algorithms to suggest items that in the past were rated highly by them. Because data collected from IPTV users cover personal information about different content they have watched or purchased, there is a serious threat to individual privacy. This data can be used for unsolicited marketing, government surveillance, profiling of usersor it can be sold by the providers when they face bankruptcy.

While in the general case the collecting of high quality profiles from users is desirable, as such recommendations can be highly beneficial for both the users and the IPTV service providers, but it is not an easy task as the price is also high: total loss of privacy while generating recommendations. On the one hand, some users are willing to reveal their whole profiles in order to get accurate referrals, but others may be concerned about the privacy implications of disclosing their profiles which can open a door for the misuse of personal data. Currently there are two options for privacy concerned users when using IPTV recommender services: first, they can refuse to enter the information they are uncomfortable about disclosing, which brings about the sparse data problem [2] for the recommendation technique, since only a subset of items have ratings scored by the user. Second, they can enter false information which decreases the accuracy of the generated recommendations, this results in a lack of acceptance of the respective services in general. As a matter of fact, an actual rating given to an item by a user produces a reasonable explanation and rank from a reliable source. Users are more likely to be willing to give more truthful data if privacy measurements are provided, or if they are assured that the data does not leave their personal devices until it is properly desensitised.

In this work, we present an enhanced middleware for collaborative privacy (*EMCP*) that allows creating serendipity recommendations without breaching user privacy. *EMCP* employs a set of mechanisms to allow users to share their data among each other in the network to attain collaborative

privacy. The users' cooperation is needed not only to protect their privacy but also to allow the service to run properly. This approach preserves the aggregates in the dataset to maximize the usability of information in order to accurately predicate ratings for items that have not previously been viewed by the target-user. Two novel mechanisms employed in *EMCP* to secure the user rating profile in the untrusted PRS with minimum loss of accuracy. The first mechanism is based on an algorithm called a Clustering Transformation Algorithm (*CTA*), for obfuscating the user rating profile before sharing it with other users in the IPTV network. It partitions the user rating profile into smaller clusters and then obfuscates each cluster in a way to preserve the distances between data points inside the same cluster. The second mechanism is based on a protocol called the Secure Recommendation (*SR*), that is build upon the Paillier scheme of homomorphic encryption in order to permit particular operations to be performed on encrypted data without need for prior decryption. This means that we can retrieve the original statistical properties without using the raw user's data. In addition, *EMCP* employs interpersonal trust between users to enhance recommendations' accuracy and preserve privacy. The enhancement in accuracy is achieved by employing trust based heuristics to propagate and spot users whom are trustworthy with respect to the target user. Moreover, trust based heuristics enhance privacy by transforming participants' data in different ways based on different trust levels to hide the raw ratings. Thus, in contrast to a single obfuscation level scenario where only one obfuscated copy is released for all users using fixed parameters for the obfuscation mechanism, now multiple differently obfuscated copies of the same data are released for different users with different trust levels. The more trusted the user, the less obfuscated the copy he can access. These different copies can be generated in various fashions. They can be jointly generated at different times upon receiving a new request from target user, or on demand. The latter case gives users the maximum flexibility.

In rest of this work, we will generically refer to news programs, movies and video on-demand content as Items. This paper is organised as follows. In Section II, related works are described. Section III introduces an IPTV network scenario hosting our private recommender service. The proposed solution based on *EMCP* is introduced in Section IV. In Section V, the two-stage concealment process is described in detail. Proof of security and correctness for the two-stage concealment process is demonstrated in Section VI. In Section VII, the results from some of the experiments on the proposed mechanisms are reported. Finally, the conclusions and recommendations for future work are given in Section VIII.

II. Related Works

The majority of the existing recommender services are based on collaborative filtering, others focus on content based filtering using EPG data [3]. Collaborative filtering techniques build users' profiles in two ways, on ratings (explicit rating procedures) or on log archives (implicit rating procedures) [4]. These procedures lead to two different approaches to collaborative filtering, rating based approaches and log based approaches. The majority of the literature addresses the problem of privacy on collaborative filtering techniques, due to it being a potential source of leakage of private information shared by the users as shown in [5]. In [6] a theoretical framework is proposed to preserve the privacy of customers and the commercial interests of merchants. Their system is a hybrid recommender system that uses secure two party protocols and public key infrastructure to achieve the desired goals. In [7, 8] a privacy preserving approach is proposed based on peer to peer techniques using users' communities, where the community will have a aggregate user profile representing the group as a whole but not individual users. Personal information will be encrypted and communication will be between individual users but not servers. Thus, the recommendations will be generated on the client side. In [9, 10] another method is suggested for privacy preserving on a centralised recommender systems by adding uncertainty to the data by using a randomised perturbation technique while attempting to make sure that the necessary statistical aggregates such as the mean do not greatly get disturbed. Hence, the server has no knowledge about the true values of the individual rating profiles for each user. They demonstrate that this method does not essentially decrease the accuracy obtained in the results. But recent research work [11, 12] pointed out that these techniques do not provide levels of privacy as was previously thought. In [12], it is pointed out that arbitrary randomisation is not safe because it is easy to breach the privacy protection it offers. They proposed random matrix based spectral filtering techniques to recover the original data from the perturbed data. Their experiments revealed that in many cases, random perturbation techniques preserve very little privacy. Similar limitations were detailed in [11]. Storing user's rating profiles on their own side and running the recommender system in a distributed manner without relying on any server is another approach proposed in [13], where the authors proposed only transmitting similarity measures over the network and keeping users' rating profiles secret on their side to preserve privacy. Although this method eliminates the main source of threat against user's privacy, it requires higher cooperation among the users to generate useful recommendations. The work in [14] stated that existing similarities are deemed useless as traditional user profiles are sparse and insufficient. Recommender systems need new ways to calculate user similarities. They utilise interpersonal trustworthiness to describe the relationship between two users. The authors in [15] show the correlation between similarity and trust and how it can elevate movie recommendation accuracy.

In this work, *EMCP* preserves the privacy of user rating profile form of the attack model presented in [16]. The attack model for data obfuscation is different from the attack model for encryption-based techniques, but no common standard has been implemented for data obfuscation. Existing attack models have primarily considered a case where the attacker correlates obfuscated data with data from other publicly-accessible databases in order to reveal the sensitive information. But the attack model presented in [16] considers a case where the attacker colludes with some users in the network to obtain some partial information about the process used to obfuscate the data and/or some of the original data items themselves. The attacker can then use this partial information to attempt to reverse engineer the entire dataset.

III. Private Recommender Service for IPTV Network Sceniro

We extend the scenario proposed in [17-21], where a private recommender service (PRS) is implemented as an external third party server and users give their rating profiles to that server in order to receive recommendations. The basic idea for a recommendation based on *EMCP* is as follows:

upon receiving a request from the target user, a group of participants is formed that is managed by an elected super-peer. Each participant obfuscates his ratings profile using a multi-level obfuscation mechanism provided by *EMCP*, such that each profile is obfuscated based on the estimated trust level with the target user, furthermore this step prevents the super-peers from knowing each participant's raw ratings. The super-peer collects these obfuscated rating profiles and computes an aggregation on it, which does not expose the individual ratings. Next, the aggregated data is encapsulated using double encryption and submitted to PRS to predicate ratings for recommended items that will be offered in the end to the target-user. The collaborative filtering task at PRS will be reduced to computing additions on aggregated data without exposing the raw data.

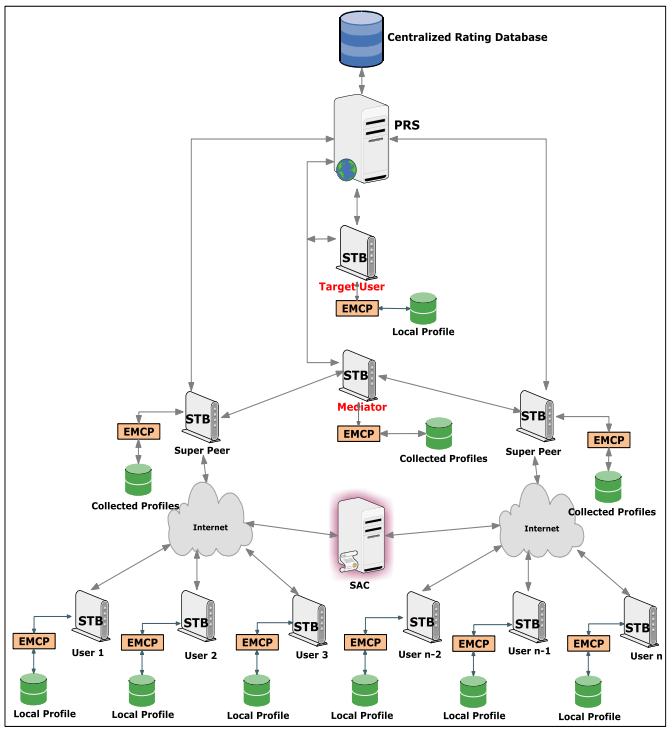


Figure 1: IPTV Network with Third Party Private Recommender Service

We do not assume the PRS to be completely malicious. This is a realistic assumption because PRS needs to accomplish some business goals and increase its revenue. Intuitively, the system privacy is high if PRS is not able to reconstruct the real ratings for users based on the information available to it. Figure (1) shows the architecture of our approach. Our solution relies on the hierarchical topology proposed in [22]; where participants are organised into peer-groups managed by super-peers. Electing super-peers is based on negotiation between the participants and security authority centre. The security authority centre (SAC) is a trusted third party responsible for generating certificates for both the target-user and mediator, and managing these certificates. In addition, SAC is responsible for making an assessment on those super-peers according to the participants' reports, and periodically updating the reputation of these super-peers. The reputation mechanisms are employed to elect suitable super-peers based on estimating values for user-satisfaction, trust level, processing capabilities and available bandwidth, further detail and information on complex reputation mechanisms can be found in [23]. When a problem with a specific super-peer occurs during the recommendation process, a participant can report it to SAC. After investigation, the assessment of the super-peer will be degraded. This will limit the chance for electing it as a super-peer in the future. On the other hand, successful recommendation processes will help upgrade the super-peer's reputation. An IPTV provider can offer certain benefits (like free content, prizes, etc.) for those participants who have a sustained success rate as a super-peer.

Our solution depends upon the set top box (STB) device at the user side. STB is an electronic appliance that connects to both the network and the home television. With the advancement of data storage technology each STB is equipped with mass storage, e.g. Cisco STB. *EMCP* components are hosted on STB; Moreover STB storage stores the user rating profile. On the other hand, PRS maintains a centralised rating database that is used to provide recommendations when the number of participants falls below a certain threshold. PRS is the third-party entity recruited by the IPTV network provider to operate recommendations by consolidating the information received from multiple sources. We alleviate the user's identity problems by using anonymous pseudonym identities for participants.

IV. Proposed Solution

In the beginning, if we want to introduce the notions of privacy and trust within our framework, we need to confirm what we mean by privacy and trust first. To define privacy and trust in our terms, we first approach the notion of privacy in following terms: "A target user who wants recommendations in a network of users, does not has to reveal raw ratings in his/her profile during the recommendations process, and other users in the network cannot learn any ratings in his/her raw profile". While in the context of this paper, trust is interpreted as "a user's expectation of another user's competency in providing ratings to reduce its uncertainty in predicating new items' ratings [24]". In our framework, the notion of privacy surrounding the disclosure of users' rating profiles and the protection of trust computation between different users are together the backbone of our solution. We apply a multi-level obfuscation mechanism that produces different copies of a participant's rating profile based on the different trust levels of the target users. The trust value is computed locally using trust engine over obfuscated users' rating profiles, and then recommendations are served using secure multi-party computation protocol. Utilising trust heuristic as input for both group formation and multi-level obfuscation has been of great importance in mitigating some of malicious insider attacks such as infesting the trust computation results. As future work, we plan to investigate miscellaneous insider attacks and strengthen our framework against them.

In the next sub-sections, we will present our proposed middleware for protecting the privacy of users' rating profiles.

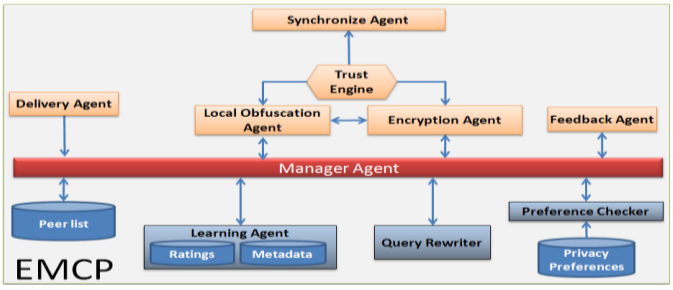


Figure 2: *EMCP* Components

Figure (2) illustrates the *EMCP* components running in the user's STB. *EMCP* consists of different co-operative agents. A learning agent captures user ratings about items explicitly or implicitly to build a rating database and meta-data database. The local obfuscation agent implements a multi-level obfuscation mechanism to achieve user privacy while sharing his/her rating profile with super-peers or PRS. The encryption agent is only invoked if the user is acting as a super-peer in the recommendation process; it executes *SR* protocol on the collected rating profiles. These mechanisms act as wrappers that obfuscate items' ratings before they are shared with any external entity. Since the database is dynamic in nature, the local obfuscation agent periodically desensitises the updated data, and then a synchronised agent forwards it to the PRS following owner permission. Thus recommendations can be made on the most recent ratings.

The recommendation process in our solution operates as follows:

1. The learning agent collects the user's ratings about different items which represent his profile. The local profile is stored on two databases, the first one is the rating database that contains (item_id, rating) and the second is the meta-data database that contains the feature vector for each item [25] (item_id, feature1, feature2, feature3). The feature vector can include genres, directors, actors and so on. Both implicit and explicit ways for information collection [26] are used to construct these two databases and maintain them. Clustering of the user's profile [27] is an extra step done by the learning agent to reduce response time for different recommendation requests.

2. As stated in [18], the target user broadcasts a message to other users in the IPTV network to request recommendations for a specific genre or category of items. Then he uses a local obfuscation agent to sanitise the rest of the items' ratings in the local profile. In order to hide the item identifiers and meta-data from other participants, the manging agent uses locality-sensitive hashing (LSH) [28] to hash these values. One interesting property for LSH is that similar items will be hashed to the same value with high probability. Super-peers and PRS are still able to perform computation on the hashed items using appropriate distance metrics like hamming distance or dice coefficient. Finally, the target user dispatches these obfuscated items' ratings along with their associated hashed values to the individual users who have decided to participate in the recommendation process. These ratings are used in the computation of trust levels at the participant side.

3. Each group of participants negotiates with SAC to select a peer with the highest reputation as a "super-peer", who will act as a communication gateway between the target user and the participants in its underlying peer group.

4. The preparation phase of *SR* protocols starts such that the target user and super-peers need to independently generate a cryptographic key. One of the super-peers will act as a mediator that has to generate an encryption key Mpk then broadcast it to the target user and all super-peers. The target user initiates the process by sending his encryption key $\varepsilon_{Mpk}(\varepsilon_{Tpk})$ to the mediator; the mediator in turn decrypts the received value to obtain Tpk. Both parties exchange their encryption keys while their decryption keys are kept private and are not shared with the other participants. Next, the mediator broadcasts Tpk to all super-peers. At the end of this phase, all super-peers hold the two encryption keys that will be used to doubly encrypt the aggregated ratings in the next phase. The target-user doubly encrypts his/her mean rating over the rated items with his/her encryption key and the mediator key. Then, he/she submits the encrypted value to PRS.

5. The process of calculating interpersonal trust with the target user is done in a decentralised fashion using the entropy definition proposed in [24] at each participant side. The entropy value becomes lower as the users' ratings are more consistent, which is similar to the definition of trust previously stated. $\forall_{j=1}^{n} T\left(u_a, u_{b_j}\right)$ is the estimated trust between the target user u_a and participant u_{b_j}, it is computed privately using the following steps:

   i. Each participant $\forall_{j=1}^{n} u_{b_j}$ determines a subset of his/her items' ratings that will be required for recommendation process. Then the participant utilises shared items rated by both of them u_a, u_{b_j} for the trust computation. Determining shared rated items is done by matching the received items' hash values from target user u_a with his/her local items' hash values.

   ii. Participant u_{b_j} invokes the trust engine to compute the trust level using equation $T\left(u_a, u_{b_j}\right) = \frac{Entropy(u_a) - Entropy\left(u_a \middle| u_{b_j}\right)}{Entropy(u_a)}$ (1)
   $= \frac{\left(1 - \frac{log\, N}{log\, ZN}\right) + \frac{1}{N\, log\, ZN}\left(\sum_{i=1}^{Z}\sum_{j=1}^{Z} n_{ij}\, log\, n_{ij} - \sum_{i=1}^{Z} n_i\, log\, n_i\right)}{1 - \frac{1}{N\, log\, ZN}\sum_{i=1}^{Z} n_i\, log\, n_i}$
   Equation (1) is an adapted formalisation of trust as proposed in [24], where Z denotes the number of states of rated values, and N is the total number of rating times. For example, if $Z=6$ and $N=20$ this means that 20 ratings are made with 1 to 6 integer valued scores. Employing entropy to select trustworthy neighbours achieves an improvement in the group formation and rating predication. Enhancement in rating predication is stemmed from trust propagation, so if $u_{b_{j=x}}$ is selected as a trustworthy user and he/she does not have ratings for the item to be predicted, a trustworthy user $u_{b_{j=y}}$ of user $u_{b_{j=x}}$ can also be used for the predication.

   iii. Each participant $\forall_{j=1}^{n} u_{b_j}$ sends his/her calculated trust value to the super-peer. The estimated trust values are forwarded to both the super-peers and PRS.

6. Each participant $\forall_{j=1}^{n} u_{b_j}$ uses their local obfuscation agent to perform a multi-level obfuscation on the items' ratings that are required in the recommendation process. Moreover the managing agent hashes their identifiers and meta-data using LSH. The level of obfuscation is determined using the trust level with the target user, and then participants submit their locally obfuscated profiles to the super-peer of their group. Secure routing protocols [29] can be utilised to hide the network identities of group members when submitting their locally obfuscated profiles to the super-peers.

7. Upon receiving the obfuscated profiles from the participants, each super-peer filters the profiles received based on the trust level of their owners, such that

$T\left(u_a, u_{b_j}\right) > \theta$ where θ is a minimum trust threshold value defined by the target user. Then, each super-peer collects the participants' pseudonyms and aggregates group profiles such that all the <hashed value, rating> elements belonging to similar items are clustered together. This allows the computing of the item's popularity curve at each super-peer. The super-peer can seamlessly interact with the PRS by posing as an end-user and has a group profile as his own profile. Each super-peer $\forall_{x=1}^{k}\, SP_x$ calculates the following intermediate values for each user in the *N*-neighbourhood of target user $\forall_{j=1}^{n} u_{b_j} \in \text{Neighbour}(u_a)$,

Then $\forall\, q = 1 \ldots T \quad \widetilde{r_{u_{b_j},q}} = r_{u_{b_j},q} - \bar{r_q}$

$$\widehat{r_{q,u_{b_j}}}^{x} = \frac{T\left(u_a,u_{b_j}\right) * \widetilde{r_{u_{b_j},q}}}{T\left(u_a,u_{b_j}\right)} \qquad (2)$$

Where $r_{u_{b_j},q}$ is the rating value of participant u_{b_j} for item q. $\bar{r_q}$ is the average rating for item q in each items' cluster. Next, each super-peer performs a double encryption on the intermediate ratings $\widehat{r_{q,u_{b_j}}}^{x}$ of each participant using the encryption key of the target user Tpk and mediator Mpk (encryption phase). Finally, the super-peer submits these ratings along with their associated hashed values to PRS, which in turn collects them to produce final referrals.

8. Upon receiving the doubly encrypted ratings $\forall_{x=1}^{k}\forall_{j=1}^{n}\varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\widehat{r_{q,u_{b_j}}}^{x}\right)\right)$ from all super-peers, PRS stores them along with their participants' pseudonyms and hashed values in the centralised rating database. The recommendation phase is performed using the additive homomorphic property of the Paillier encryption as the required computations are additions. Thus, PRS executes an additive operation on the doubly encrypted rating profiles without decrypting them so the private data of multiple super-peers can be preserved during the computation. Calculating the predicted rating for referrals is done as shown in equation (3):

$$p_{u_a,q} = \varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\overline{r_{u_a}}\right)\right) * \left(\prod\nolimits_{j=1}^{n} \varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\widehat{r_{q,u_{b_j}}}^{x}\right)\right)\right)$$

$$= \varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\overline{r_{u_a}} + \left(\sum\nolimits_{j=1}^{n} \widehat{r_{q,u_{b_j}}}^{x}\right)\right)\right) \qquad (3)$$

Notice that the result will be equal to the weighted sum of the participants' rating plus the average rating of the target user r_{u_a}. PRS uses the reblinding property of the Paillier encryption to prevent the mediator from obtaining any knowledge of $p_{u_a,q}$ values before sending them back to the target user by trying a few possible values.

9. PRS forwards the doubly encrypted referrals list along with their predicated ratings to the mediator that in turn decrypts it and forwards the output to the target user. The target-user in turn decrypts the list to obtain the final output because he/she holds the final decryption key. Optionally, the target-user publishes the final list to other participants in the recommendation process. Finally, each participant reports their score about the elected super-peer of his group and target-user to SAC, which helps determine the reputation of each entity involved in the referral's generation.

V. Proposed Two-Stage Concealment Process

In the next subsections, we present a two-stage concealment process used in *EMCP* to disguise the user rating profile in a way that secure the user's ratings in the untrusted PRS with minimum loss of accuracy. In our framework, each user has two datasets representing his/her profile. A local profile: represents the actual ratings of the user for different items; it is stored on his STB. Each user disguises this local profile before sending it to super-peer. A centralised profile: this is the output of the two-stage concealment process where the user gets recommendation directly from the PRS based on previously collected profiles. We perform experiments on real datasets to illustrate the applicability of our mechanisms and the privacy and accuracy levels achieved by using them.

A. Cryptigrpahy Tools

We employ a homomorphic encryption scheme to preserve the privacy of ratings collected by super-peers. Moreover, homomorphic encryption possesses specific properties that permit the computation of linear combinations of encrypted data without need for prior decryption. Formally, an encryption schema $\varepsilon_{pk}(.)$ denotes the encryption function with encryption key pk and $D_{sk}(.)$ denoted by the decryption function with decryption key sk. Additive homomorphic cryptosystem possess the following properties:

1. Given the encryption of plaintexts m_1 and m_2, $\varepsilon_{pk}(m_1)$ and $\varepsilon_{pk}(m_2)$. The sum $m_1 + m_2$ can be directly computed as $\varepsilon_{pk}(m_1 + m_2) = \varepsilon_{pk}(m_1) * \varepsilon_{pk}(m_2)$.
2. Given a constant k and the encryption of m_1 , $\varepsilon_{pk}(m_1)$. The multiplication of k with the plaintext m_1 can be directly computed as $\varepsilon_{pk}(k.m_1)$= $\varepsilon_{pk}(m_1)^k$.

In designing SR protocol we used the Paillier cryptosystem as it provides strong security due to the use of randomised encryption, so an adversary cannot even see whether two encryptions correspond to the same text. We will briefly present the Paillier cryptosystem, however a more detailed description can be found in [30].

Key Generation

In this step, two large primes p and q are chosen randomly where $p < q$ and $p \nmid q - 1$. The encryption key pk is set to N where $N = p.q$, and the decryption key sk is set to (λ, N) where $\lambda = lCM(p-1, q-1)$.

Encryption

Given n, $g \in \mathbb{Z}_{N^2}^{*}$ is a generator whose order divides N, plaintext m and a random number $r \in [1, \ldots., N-1]$. The encryption of the message $m \in \mathbb{Z}_n$:$\varepsilon_{pk}(m) = g^m.r^N mod\ N^2$. For any encrypted message, a different encryption can be computed by multiplying it with a random blinding factor r^N.

Decryption

Given N, the cipher-text $c = \varepsilon_{pk}(m)$, and the decryption is as follows $m = \frac{(c^{\lambda} mod\ N^2)-1}{N}\lambda^{-1}$ where λ^{-1} is the inverse of λ in module N.

B. Local Obfuscation using Clustering Transformation Algorithm (CTA)

We propose a novel algorithm for obfuscating the user rating profile before sending it to the super-peer. This algorithm is called *CTA*, which has been designed especially for the sparse data problem we have here. Moreover the algorithm permits multi-level obfuscation based on trust level.

We noted that the available anonymisation algorithms increase data distortion, as they perform single obfuscation levels for all participants and release one obfuscated copy for all of them, and as a result an inaccurate recommendation model could be constructed. Maintaining utility and privacy for profiles seem to be contradictory goals. *CTA* partitions the user profile into smaller clusters and then pre-process each cluster such that the distances inside the same cluster will be maintained in its obfuscated version. We use the local learning analysis (*LLA*) clustering method proposed in [31] to partition the dataset. After completing the partitioning, we embed each cluster into a random dimension space so the sensitive ratings will be protected. Then the resulting cluster will be rotated randomly. In such a way, *CTA* obfuscates the data inside the user profile while preserving the distances between the data points to provide accurate results when performing recommendations. The output of our obfuscation algorithm should satisfy two requirements:

- Reconstructing the original profile from the obfuscated profile should be difficult, in order to preserve privacy.
- Preserving the distances of the data to achieve accurate results for the recommendations.

Our algorithm consists of the following steps:

1. The user ratings is stored in STB as dataset D consisting of c rows, where each row is a sequence of X attributes where $X = x_1\ x_2\ x_3 \dots\dots\dots\ x_n$.
2. The dataset D is portioned vertically into $D_1\ D_2\ D_2 \dots\dots D_m$ subsets of length L, if n/L is not perfectly divisible then *CTA* randomly selects attributes already assigned to any subset and joins them to the attributes of the incomplete subsets.
3. Cluster each subset $\forall_{j=1}^{m} D_j$ Using *LLA* algorithm, which results in K clusters $D_j = C_{j1}, C_{j2}, C_{j3} \dots . C_{jk}$ for each subset. Note that *LLA* uses the Gaussian influence function as the similarity measure. The influence function between two data point x_i and x_j is given by:

$$f_{Gauss}^{x_i}(x_j) = e^{-\frac{d(x_i,x_j)^2}{2\sigma^2}} \qquad (4)$$

The field function for a candidate core-point selection is given by:

$$f_{Gauss}^{D}(x_j) = \sum_{s=1}^{k} e^{-\frac{d(x_j,x_{is})^2}{2\sigma^2}} \qquad (5)$$

So every point in the original dataset D falls exactly into one cluster. The aim of this step is to increase the privacy level of the transformation process and make reconstruction attacks difficult.

4. *CTA* generates two sets for each cluster $\forall_{i=1}^{k} C_{ji}$ in the subset D_j these are H_{ji} and O_{ji}. Where H_{ji} is the set of points with the highest values for the field function and O_{ji} are the rest of points in C_{ji}. For each point $x_{1i} \in H_{ji}$ constructs a weighted graph Γ_i that contains its k-nearest neighbours in O_{ji}, each edge $e \in \Gamma_i$ has a weight equal to $f_{Gauss}^{b_{1i}}(x_{1i})$.
5. Estimate the geodesic distances by computing the shortest distance between the two points in graph Γ_i using Dijkstra or Floyd algorithms, and then build a distance matrix $D_{\Gamma_i} = \{f_{Gauss}^{b_i}(x_i)\}$.
6. Based on D_{Γ_i}, we find a d-dim embedding space $C_{ji}^{'}$ using classical *MDS* [32] as follows:
   - Calculate the matrix of squared distances $S = D_{\Gamma_i}^2$ and the centring matrix $H = 1 - 1/N\ ee^T$
   - The characteristic vectors are chosen to minimise $E = \left\|\tau(D_{\Gamma_i}) - \tau(D_d)\right\|_{L^2}$, where $\tau(D_d)$ is the distance matrix for the *d-dim* embedding space, and τ converts distances to inner products $\tau = -HSH/2$.
   - Trust level values are divided into a number of intervals defined by the user, such that each interval is associated with a *d*-dim value. *CTA* chooses a value for *d*-dim according to the trust level associated with the target user.
7. For each cluster $\forall_{j=1}^{m} \forall_{i=1}^{k} C_{ji}^{'}$, *CTA* randomly selects two attributes x_a and x_b to perform rotation perturbation on the selected attributes $R(x_a, x_b)$ using the transformation matrix M_j^{θ} setup by the user for each cluster, using a range of angles defined in advance by the user.
8. Repeating steps 4–7 for all clusters in $\forall_{j=1}^{m} D_j$ to obtain the obfuscated portion $D_j^{'}$. Finally, the obfuscated dataset is obtained by $D^{'} = \cup_{j=1}^{n} D_j^{'}$.

C. Secure Recommendation Protocol (SR)

We proposed a protocol that enables PRS to calculate predicted ratings from the doubly encrypted profiles. We called this protocol the secure recommendation protocol (*SR*). *SR* consists of three phases: preparation phase, encryption phase and recommendation phase, as stated in section IV. Employing a multi-level obfuscation mechanism on local profiles based on the estimated trust level with target-user ensures the participants' privacy as only the local profile leaves the participant's device desensitised. Obfuscating local profiles permits super-peers to perform intermediate computation on their obfuscated shares of ratings without need for extra SMC protocols or large encryption keys. Moreover, obfuscation requires lees computational and communication resources than required for encryption. The final recommendation phase is carried out in the form of the secure addition of encrypted ratings profile received from super-peers.

1. Preparation phase: generation of cryptographic key pair by mediator and target user
   For mediator
   Generate encryption key Mpk
   Broadcast Mpk to all super-peers and target user
   Broadcast Tpk to all super-peers
   End for
   For Target-user
   Generate encryption key Tpk
   Send $\varepsilon_{Mpk}(\varepsilon_{Tpk})$ to mediator
   End for
2. Encryption phase: Each super-peer encrypts his/her aggregated ratings profile with both encryption keys
   For each participant $\forall_{j=1}^{n} u_{b_j}$ do
   Extracts his/her rating for requested item $r_{u_{b_j},q}$
   Calculates the trust level with target user $T\left(u_a, u_{b_j}\right)$
   Perform multi-level obfuscation on $\forall_T^1 r_{u_{b_j},q}$
   Send $r_{u_{b_j},q}$, $T\left(u_a, u_{b_j}\right)$ to the super-peer in his group
   End for
   For each super-peer $\forall_{x=1}^{k} SP_x$ do

Calculates $\widehat{r_{q,u_{b_j}}}^{x} = \frac{T(u_a,u_{b_j})*\widehat{r_{u_{b_j}}}^{q}}{T(u_a,u_{b_j})}$

Send $\widehat{r_{q,u_{b_j}}}^{x'} = \varepsilon_{MpK}\left(\varepsilon_{TpK}\left(\widehat{r_{q,u_{b_j}}}^{x}\right)\right)$ to PRS

End for

3. Recommendation phase: PRS generates referrals by accumulating the received shares in order to obtain a predicated rating for each referred item. Detailed steps can be stated as follows:

PRS receives $\widehat{r_{1,u_{b_1}}}', \widehat{r_{1,u_{b_1}}}', \ldots, \widehat{r_{2u_{b_1}}}', \widehat{r_{2u_{b_1}}}', \ldots, \widehat{r_{Tu_{b_j}}}'$ such that $\widehat{r_{qu_{b_j}}}'$ is the doubly encrypted rating for item $q \in \{1, \ldots \ldots T\}$ by user $u_{b_j}\left(\forall_{j=1}^{n} u_{b_j} \mid T\left(u_a, u_{b_j}\right) > \theta\right)$ from super-peer $SP_x(\forall_{x=1}^{k} SP_x)$

For each item $q = 1$ to T do

PRS Calculates $\forall_{x=1}^{k}\, p_{u_a,q} = \varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\overline{r_{u_a}}\right)\right) * \left(\prod_{j=1}^{n} \varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\widehat{r_{q,u_{b_j}}}^{x}\right)\right)\right) \forall_{j=1}^{n} u_{b_j}$

End for

PRS sends the list of items and their predicated ratings $\{(item_1, p_{u_a,1}), (item_2, p_{u_a,2}), \ldots (item_T, p_{u_a,T})\}$ to the mediator. Next, the mediator will decrypt the received list and sends it to the target user. The target-user is able to find the final output because it holds the final decryption key. The target user filters out the received list on his device by removing items with a low predicated rating and items already viewed.

VI. Proof of Security and Correctness

The proof of security for *SR* protocol depends on how much information is leaked during the execution of the recommendation phase. At the same time, our *SR* protocol should output accurate results.

Theorem 1: additive operation performed by PRS in *SR* protocol is correct and accurate without the need of decryption keys.

Proof: based on the first property of additive homomorphic cryptosystem, we can determine that additive operations for doubly encrypted data are correct as follows: given the encryptions $\varepsilon_{pk1}(m_1) = a$ and $\varepsilon_{pk1}(m_2) = b$ where $\forall m_1, m_2 \in \mathbb{Z}_n$, given encryption key $pk2$ $\varepsilon_{pk2}\left(\varepsilon_{pk1}(m_1)\right).\ \varepsilon_{pk2}\left(\varepsilon_{pk1}(m_2)\right)\ mod\ N^2$ = $\varepsilon_{pk2}(a).\ \varepsilon_{pk2}(b) = (g^a r_1^N).(g^b r_2^N)\ mod\ N^2 =$ $g^{a+b}(r_1 r_2)^N\ mod\ N^2 = \varepsilon_{pk2}\left(\varepsilon_{pk1}(m_1 + m_2)\right)\ mod\ N^2$ Based on that, the PRS does not require any decryption key in order to aggregate all encrypted data.

Theorem 2: *SR* protocol computes predicated ratings for each referred item based on similar user ratings without revealing extra information to any party.

Proof: Since each participant obfuscates items' ratings and hashes their meta-data prior to submitting them to the super-peer. Moreover, each super-peer encrypts the aggregated profiles with the encryption keys of the target user and mediator. No single party is able to decrypt the encrypted profiles. In our two-stage concealment process, the super-peer aggregates all obfuscated profiles then performs intermediate-computations on the obfuscated ratings for each item without having to know their real ratings. No party can see the extra information during the execution of the *SR* protocol, except the target user at the end of the protocol. As for participants, they participate in the recommendation process without knowing other participants' identity, since not all the participants have the same super-peer nor do they have direct communication with each other. The local profile is secured and can only be viewed by its owner before applying the multi-level obfuscation mechanism. In addition, employing reputation techniques to select super-peers with a high success rate in previous recommendation processes ensures the selection of reliable peers that will perform the required phases. PRS cannot see the received profiles as none of the decryption keys are known. Furthermore, the decryption process requires both decryption keys stored at the target-user and mediator. After PRS generates the final referrals list, PRS submits it to the mediator which in turn perform the first decryption process. After the first decryption, the mediator is not able to see the final result because the final decryption key is held by the target user.

Theorem 3: Assuming that all parties follow the protocol, *SR* protocol can correctly compute the predicated rating for each referred item.

Proof: When each super-peer encrypts his aggregated profiles with both encryption keys $\varepsilon_{MpK}\left(\varepsilon_{TpK}\left(\widehat{r_{q,u_{b_j}}}^{x}\right)\right)$. PRS performs the additive operation on doubly encrypted profiles based on Paillier's homomorphic cryptosystem as follows:

$$p_{u_a,q} = \varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\overline{r_{u_a}}\right)\right) + \left(\varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\widehat{r_{q,1}}\right)\right) + \varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\widehat{r_{q,2}}\right)\right) + \varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\widehat{r_{q,3}}\right)\right) + \cdots + \varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\widehat{r_{q,u_{b_j}}}\right)\right)\right)$$

$$p_{u_a,q} = \varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\overline{r_{u_a}} + \left(\sum_{j=1}^{n} \widehat{r_{q,u_{b_j}}}^{x}\right)\right)\right)$$

$$p_{u_a,q} = \varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\overline{r_{u_a}}\right)\right) * \left(\prod_{j=1}^{n} \varepsilon_{Mpk}\left(\varepsilon_{Tpk}\left(\widehat{r_{q,u_{b_j}}}^{x}\right)\right)\right) \quad (6)$$

After the first decryption by the mediator, we have

$$p_{u_a,q} = \varepsilon_{Tpk}\left(\overline{r_{u_a}}\right) * \left(\prod_{j=1}^{n} \varepsilon_{Tpk}\left(\widehat{r_{q,u_{b_j}}}^{x}\right)\right) \quad (7)$$

When the target-user performs the final decryption, he will obtain the final predicated rating as in equation (8)

$$p_{u_a,q} = \overline{r_{u_a}} * \left(\prod_{j=1}^{n} \widehat{r_{q,u_{b_j}}}^{x}\right) \quad (8)$$

So the result from *SR* protocol is correct.

VII. Experiments

In this section, we describe the implementation of our proposed solution. The experiments are run on an Intel® Core 2 Duo™ 2.4 GHz processor with 2 GB Ram. We used MySQL as data storage. The proposed two-stage concealment process is implemented in C++. We used a message passing interface (MPI) for the distributed memory implementation of the *SR* protocol to mimic a distributed reliable network of peers. The experiments presented here were conducted using the Jester dataset provided by Goldberg from UC Berkley [33]. The dataset contains 4.1 million ratings on jokes using a real value between (-10 and +10) of 100 jokes from 73,412 users. The data in our experiments consists of ratings for 36 or more items by 23,500 users. We evaluated the proposed solution from two different aspects: privacy achieved and accuracy of results. We used the mean absolute error (MAE) metric proposed in [34]. MAE is one of most famous metrics for

recommendation quality. We can define it as follows: given a user predicated ratings set $p = \{p_1, p_2, p_3 \dots p_N\}$ and the corresponding real ratings set $r = \{r_1, r_2, r_3 \dots . r_N\}$, MAE is:

$$MAE = \sum_{i=1}^{N} |p_i - r_i| / N \qquad (9)$$

MAE measures the predication verity between the predicated ratings and the real ratings, so a smaller MAE means better recommendations provided by PRS. To measure the privacy or distortion level achieved using our mechanism, we used the variation of information metric VI [35] to estimate data error. VI is:

$$VI = H(p) + H(r) - 2I(p,r) \qquad (10)$$

Here $H(p)$ is entropy of p, r and $I(p,r)$ is the mutual information between p and r. A higher value of VI means a larger distortion between the obfuscated and original dataset, which means a higher level of privacy.

The experiments involve dividing the data set into a training set and testing set. The training set is obfuscated then used as a database for PRS. Each rating record in the testing set is divided into rated items t_i and unrated items r_i. The set t is presented to the PRS for making predication p_i for the unrated items r_i. For the representation process of the trust calculation, we add the default value 0 for items not rated. In our dataset, the first column in the raw data store how many items are rated by the user, which is necessary for the trust estimation process. We fix the number of super-peers to 3, as described earlier they will be responsible for aggregating the data of 23,496 participants. We assume the trust level for all participants to be above the minimum trust threshold θ that is required for the inclusion in the prediction process. The recommendation process can be initiated by any user that will act as the target-user for the referrals list. The trust level between participants and target-user is calculated locally on their STB devices.

We used our *SR* protocol to predict referred items' ratings based on the weighted ratings for each participant. First we want to measure the encryption performance in *SR*, so we collect all 23,496 records on one super-peer and doubly encrypt the aggregated data with a different encryption key length of 128, 256, 512, 1024 and 2048 referring to figure (3). With 128 bits, the time elapsed by the encryption of 23,496 records is about 3120 ms and 4230 for 256 bits, 5814 for 512 bits 8164 ms for 1024 bits and 12,241ms for 2048 bits respectively. As we can see, this result presents an exponential cost in time while doubling the encryption key's length.

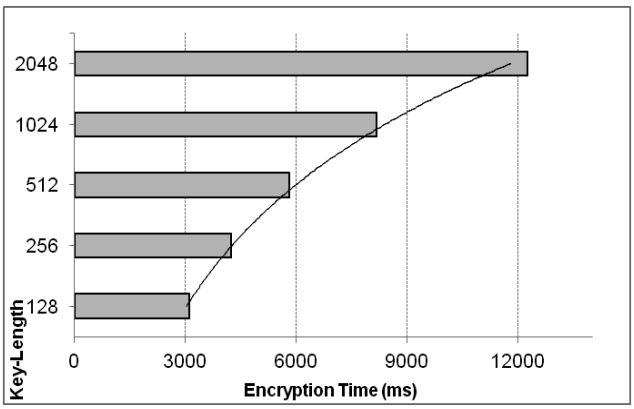


Figure 3: Encryption Time versus Key Length.

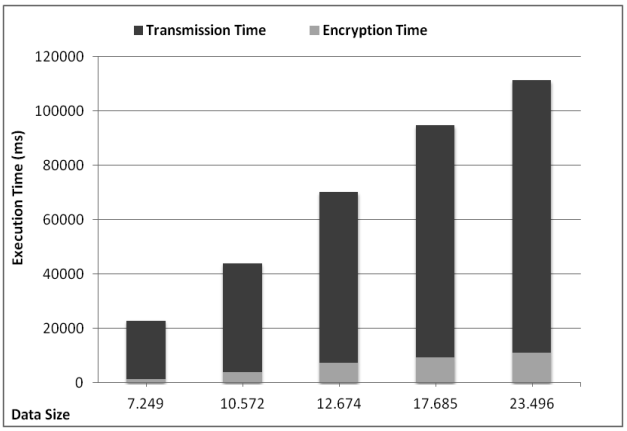


Figure 4: Execution Time for Different Data Size

In the second experiment, we vary the minimum trust threshold to obtain a different number of participants' records in the recommendation process, then we run the SR protocol on these aggregated records in sizes of 7249, 10,572, 12,674, 17,685, and 23,496. As shown in figure (4), we can see that both encryption time and transmission time grow linearly while we enlarge the data size. Moreover, we can see that the execution time is directly proportional to the data size, and transmission time dominates this increase in total cost of execution time.

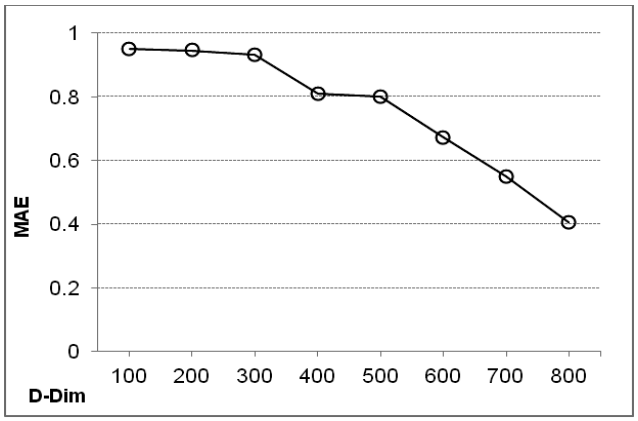


Figure 5: Accuracy of Recommendations for Obfucated Dataset using *CTA*

To evaluate the accuracy of the *CTA* algorithm with respect to different number of dimensions in the rating profile, we control *d-dim* parameters of *CTA* to vary the number of dimensions during the evaluation. Figure (5) shows the performance of recommendations of locally obfuscated data, as shown in the accuracy of recommendations based on the fact that the obfuscated data is low when the dimension is low. But at a certain number of dimensions (500), the accuracy of the recommendations of the obfuscated data is nearly equal to the accuracy obtained by using the original data.

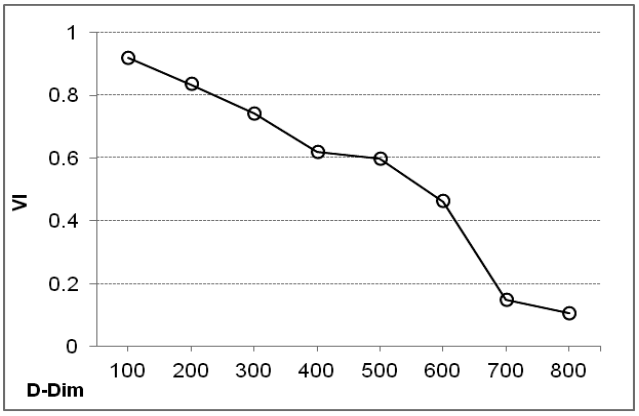


Figure 6: Privacy Levels for the Obfucated Dataset using *CTA*

In the second experiment performed on the *CTA* algorithm, we examined the effect of *d-dim* on VI values. As shown in figure (6), VI values decrease with respect to the increase in *d-dim* values for the rating profile. *d-dim* is the key element for privacy level where the smaller the *d-dim* value, the higher the VI values (privacy level) of *CTA*. However, clearly the highest privacy is at *d-dim*=100. There is a noticeable drop of VI values when we change *d-dim* from 300 to 600. A *d-dim* value of 400 is considered as a critical point for privacy. Note that the rotation transformation adds an extra privacy layer to the data and at the same time maintains the distance between data points to enable PRS to build accurate recommendation models.

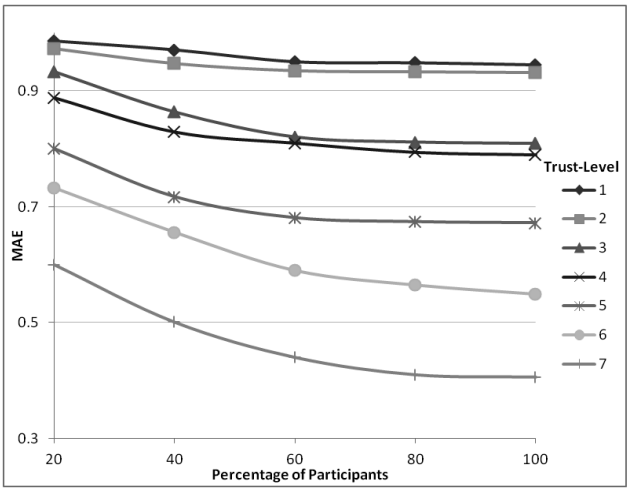


Figure 7: MAE Values with Different Percentage of Participants.

In the final experiment, we want to measure the impact of varying the trust level and number of participants on the accuracy of the recommendations. We simulate a general case where the number of users was fixed at 23,496. Then we assign a different number of participants to a certain recommendation request, and gradually increase the percentage of users who joined the request from 10% to 100%. We fixed the trust level for the target user with the participants for each simulation (trust level has an impact on *CTA* parameters), then we measured the MAE for the results. As shown in figure (7), the MAE value occurs at approximately 60% of the participants with high trust levels are close to the same MAE value for all users. Our conclusion is that a low percentage of participants employing multi-level obfuscation with ahigh trust level can obtain MAE values close to the original MAE value obtained from running the recommendation process for all users. As a result, the target user does not need to broadcast the request to the full IPTV network to obtain accurate results, but he can employ multicast for trusty users stored in his peer list to reduce the load on the network traffic. To illustrate the decrement of MAE values for predications based on diverse percentages of participants and trust levels, we calculated and plotted figure (7). This verifies our conclusion that in our case the MAE approximately converges to the MAE obtained using all the users.

VIII. CONCLUSION AND FUTURE WOK

In this paper, we presented our ongoing work on building an enhanced middleware for collaborative privacy in IPTV recommender services. We gave a brief overview of *EMCP* architecture, the recommendations process with application to IPTV. We presented a novel two-stage concealment process that provides complete privacy control to participants over their ratings profiles. The concealment process utilises hierarchical topology, where participants are organised into groups, from which super-peers are elected based on their reputation. Super-peers aggregate the results obtained from underlying participants and then encapsulate intermediate values computed on these aggregated data and then send them to PRS. A multi-level obfuscation mechanism is used in the course of participant data collection, while the *SR* protocol is used to protect the privacy of collaborative filtering by distributing the participants' data between multiple super-peers and only exchanging a subset of the aggregated ratings which is useful for the recommendations. We tested the performance of the proposed mechanisms on a real dataset. We evaluated how the overall accuracy of the recommendations depends on the number and trust level of participants. The experimental and analysis results show that privacy increases under the proposed middleware without hampering the accuracy of the recommendations. In particular, the mean absolute error can be reduced with proper tuning of the multi-level obfuscation parameters for a large number of participants. Moreover, utilising trust levels for multi-level obfuscation is an optimisation to maintain the utility of the rating profiles. Thus adding the proposed middleware does not severely affect the accuracy of the recommendations based on collaborative filtering techniques.

We realised that there are many challenges in building a privacy enhanced middleware for a recommender service. As a result, we focused on middleware in a collaborative privacy scenario. A future research agenda will include utilising game theory to better formulate user groups, sequential profile release and its impact on privacy. We will consider reducing transmission time and the load on the network traffic by adding a secure filtering phase to the *SR* protocol that will allow PRS to exclude items with low predicated rating from the final referrals list. Furthermore it is intended to strengthen our middleware against shilling attacks, extend our scheme to be directed towards multi-dimensional trust propagation and distributed collaborative filtering techniques in a p2p environment. Moreover, we need to investigate weighted features vector methods and its impact on released ratings. Such that the participant not only obfuscates his items' ratings based on the trust level of the target-user, but he can also express specific items to be diversely obfuscated with each trust level. We need to perform extensive experiments on other real datasets from the UCI repository and compare our

performance with other techniques proposed in the literature. Finally, we need to consider different data partitioning techniques as well as identify potential threats and add some protocols to ensure the privacy of the data against those threats.

ACKNOWLEDGMENT

This work has received support from the Higher Education Authority in Ireland under the PRTLI Cycle 4 Programme, in the FutureComm Project (Serving Society: Management of Future Communications Networks and Services).

REFERENCES

[1] K. Kawazoe, *et al.*, "Platform application technology using the next generation network," *NTT2007*, 2007.

[2] Z. Huang, *et al.*, "Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering," *ACM Trans. Inf. Syst.*, vol. 22, pp. 116-142, 2004.

[3] L. Ardissono, *et al.*, *Personalized digital television: Targeting programs to individual viewers (Human-Computer Interaction Series, 6)*: Kluwer Academic Publishers, 2004.

[4] M. d. Gemmis, *et al.*, "Preference learning in recommender systems," presented at the *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD)*, Slovenia, 2009.

[5] F. McSherry and I. Mironov, "Differentially private recommender systems: building privacy into the net," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Paris, France, 2009.

[6] A. Esma, "Experimental demonstration of a hybrid privacy-preserving recommender system," *Third International Conference on Availability Reliability and Security*, 2008, pp. 161-170.

[7] J. Canny, "Collaborative filtering with privacy via factor analysis," in Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Tampere, Finland, 2002.

[8] J. Canny, "Collaborative filtering with privacy," in *Proceedings of the 2002 IEEE Symposium on Security and Privacy*, 2002.

[9] H. Polat and W. Du, "Privacy-preserving collaborative filtering using randomized perturbation techniques," in *Proceedings of the Third IEEE International Conference on Data Mining*, 2003.

[10] H. Polat and W. Du, "SVD-based collaborative filtering with privacy," in *Proceedings of the 2005 ACM Symposium on Applied Computing*, Santa Fe, New Mexico, 2005.

[11] Z. Huang, *et al.*, "Deriving private information from randomized data," in *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*, Baltimore, Maryland, 2005.

[12] H. Kargupta, *et al.*, "On the privacy preserving properties of random data perturbation techniques," in *Proceedings of the Third IEEE International Conference on Data Mining*, 2003.

[13] B. N. Miller, *et al.*, "PocketLens: Toward a personal recommender system," *ACM Trans. Inf. Syst.*, vol. 22, pp. 437-476, 2004.

[14] C.-N. Ziegler, *et al.*, "Improving recommendation lists through topic diversification," in *Proceedings of the 14th International Conference on World Wide Web*, Chiba, Japan, 2005.

[15] J. Golbeck and J. Hendler, "FilmTrust: movie recommendations using trust in web-based social networks," in *3rd IEEE Consumer Communications and Networking Conference, CCNC 2006*, 2006, pp. 282-286.

[16] R. Parameswaran and D. M. Blough, "Privacy preserving data obfuscation for inherently clustered data," *Int. J. Inf. Comput. Secur.*, vol. 2, pp. 4-26, 2008.

[17] A. M. Elmisery and D. Botvich, "An agent based middleware for privacy aware recommender systems in IPTV networks," in *Intelligent Decision Technologies*. vol. 10, J. Watada, *et al.*, Eds., ed: Springer Berlin Heidelberg, 2011, pp. 821-832.

[18] A. Elmisery and D. Botvich, "Private recommendation service for IPTV system," in *12th IFIP/IEEE International Symposium on Integrated Network Management*, Dublin, Ireland, 2011.

[19] A. Elmisery and D. Botvich, "Agent based middleware for maintaining user privacy in IPTV recommender services," in *3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems*, Aalborg, Denmark, 2011.

[20] A. Elmisery and D. Botvich, "Privacy aware obfuscation middleware for mobile jukebox recommender services," in *11th IFIP Conference on e-Business, e-Service, e-Society*, Kaunas, Lithuania, 2011.

[21] A. Elmisery and D. Botvich, "Privacy aware recommender service for IPTV networks," in *5th FTRA/IEEE International Conference on Multimedia and Ubiquitous Engineering*, Crete, Greece, 2011.

[22] W. Nejdl, *et al.*, "Super-peer-based routing and clustering strategies for RDF-based peer-to-peer networks," in *Proceedings of the 12th International Conference on World Wide Web*, Budapest, Hungary, 2003.

[23] J. Carbo, *et al.*, "Trust management through fuzzy reputation. Int," *Journal in Cooperative Information Systems*, vol. 12, p. 135—155, 2002.

[24] H. D. Kim, "Applying consistency-based trust definition to collaborative filtering," *KSII Transactions on Internet and Information Systems*, vol. 3, pp. 366-374, 2009.

[25] A. Elmisery and D. Botvich, "Agent based middleware for private data mashup in iptv recommender services," in *16th IEEE International Workshop on Computer Aided Modeling, Analysis and Design of Communication Links and Networks*, Kyoto, Japan, 2011.

[26] D. Kelly and J. Teevan, "Implicit feedback for inferring user preference: a bibliography," *SIGIR Forum*, vol. 37, pp. 18-28, 2003.

[27] Y. Ye, *et al.*, "A comparative study of feature weighting methods for document co-clustering," *International Journal of Information Technology, Communications and Convergence*, vol. 1, pp. 206-220, 2010.

[28] P. Indyk and R. Motwani, "Approximate nearest neighbors: towards removing the curse of dimensionality," in *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, Dallas, Texas, United States, 1998.

[29] M. Imani, *et al.*, "Security enhanced routing protocol for ad hoc networks," *Journal of Convergence*, vol. 1, pp. 43-48, 2010.

[30] P. Paillier, "Public-key crvptosvstems based on composite degree residuosity classes," EUROCRYPT 1999, pp. 223-238..

[31] A. Elmisery and F. Huaiguo, "Privacy preserving distributed learning clustering of healthcare data using cryptography protocols," in *34th IEEE Annual International Computer Software and Applications Workshops*, Seoul, South Korea, 2010.

[32] I. Borg and P. J. F. Groenen, *Modern multidimensional scaling: theory and applications (Springer Series in Statistics)*: Springer, 2005.

[33] D. Gupta, *et al.*, "Jester 2.0 (poster abstract): Evaluation of an new linear time collaborative filtering algorithm," in *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Berkeley, California, United States, 1999.

[34] J. L. Herlocker, *et al.*, "Evaluating collaborative filtering recommender systems," *ACM Trans. Inf. Syst.*, vol. 22, pp. 5-53, 2004.

[35] C. Kingsford, "Information theory notes," 2009.

Appendix B: IPTV Recommender Service Scenario

Article VII

Privacy Aware Recommender Service using Multi-agent Middleware– an IPTV Network Scenario

Ahmed M. Elmisery, Dmitri Botvich

In the International Journal of Computing and Informatics (Informatica), Volume 36, Issue 1, March 2012.

Informatica **36** (2012) 21–36

Privacy Aware Recommender Service using Multi-agent Middleware – An IPTV Network Scenario

Ahmed M. Elmisery and Dmitri Botvich
Telecommunications Software & Systems Group-TSSG
Waterford Institute of Technology-WIT, Co. Waterford, Ireland
E-mail: ahmedmohmed2001@gmail.com

Keywords: clustering, IPTV network, recommendation-services, multi-agent, secure multiparty computation

Received: October 28, 2011

IPTV service providers are starting to realize the significant value of recommender services in attracting and satisfying customers as they offer added values e.g. by delivering suitable personalized contents according to customers personal interests in a seamless way, increase content sales and gain competitive advantage over other competitors. However the current implementations of recommender services are mostly centralized combined with collecting data from multiple users that cover personal preferences about different contents they watched or purchased. These profiles are stored at third-party providers that might be operating under different legal jurisdictions related to data privacy laws rather than the ones applied where the service is consumed. From privacy perspective, so far they are all based on either a trusted third party model or on some generalization model. In this work, we address the issue of maintaining users' privacy when using third-party recommender services and introduce a framework for Private Recommender Service (PRS) based on Enhanced Middleware for Collaborative Privacy (EMCP) running at user side. In our framework, PRS uses platform for privacy preferences (P3P) policies for specifying their data usage practices. While EMCP allows the users to use P3P policies exchange language (APPEL) for specifying their privacy preferences for the data extracted from their profiles. Moreover, EMCP executes a two-stage concealment process on the extracted data which utilize trust mechanism to augment the recommendation's accuracy and privacy. In such case, the users have a complete control over the privacy level of their profiles and they can submit their preferences in an obfuscated form without revealing any information about their data, the further computation of recommendation proceeds over the obfuscated data using secure multi-party computation protocol. We also provide an IPTV network scenario and experimentation results. Our results and analysis shows that our two-stage concealment process not only protect the users' privacy, but also can maintain the recommendation accuracy.

Povzetek: Članek obravnava priporočanje vsebin uporabnikom televizije IP, ki spoštuje uporabnikovo zasebnost.

1 Introduction

Internet Protocol Television (IPTV) is a video service providing IP broadcasts and video on demand (VOD) over a broadband IP content delivery network (CDN) specialized in video services. The IPTV user has access to myriads of video content spanning IP Broadcast and VOD [1]. In this context, it is difficult for them to find content that matches their preferences from the huge amount of video content available. In order to attract and satisfy these users, IPTV service providers employ recommendation services to increase their revenues and offer added value to their patrons. In the same time, Recommender services can improve the overall performance of the current IPTV network by building up an overlay to increase content availability, prioritization and distribution.

Recommender service offers referrals to users by building up users' profiles (explicit or implicit) based on their past ratings, behaviour, purchase history or demographic information. In the context of this work, a profile is a list comprises the video contents the user has watched or purchased combined with their meta-data extracted from the content provider (i.e. genres, directors, actors and so on) and the ratings the user gave to these contents. Maintaining the quality of offered referrals and quickly react to problems raised from merging data from different sources requires a lot of expertise, and not all IPTV providers have the ability to construct and interpret recommendation models. Therefore, there is a market for specialized firms on users' profiles storage and analysis. But there are some challenges face this business model, such as security and privacy. Because collected data from users cover personal information about different contents they have watched or purchased and these profiles might be stored at third-party providers that might be operating under different legal jurisdictions related to data privacy laws rather than the ones applied where the service is consumed. This is a serious threat to individual privacy since this data can be used for unsolicited marketing,

government surveillance, profiling of users, misused, furthermore it can be sold by providers when they face bankruptcy. As a matter of fact, users are more likely willing to give more truthful preferences if privacy measurements are provided or if they assured that the data does not leave their personal devices until it is properly desensitised.

The organization for economic co-operation and development (OECD) [2] formulated sets of principles for fair information practice that can be considered as the base for privacy laws. These principles allow the users to control the data they provide for recommender services operating at remote sites, they can be described as follows:

1. Collection limitation: data collection and usage for a recommender service should be limited only to the data it requires to offer appropriate service.
2. Data quality: data should be used only for the relevant purposes for which it is collected.
3. Purpose specification: data controllers should specify up front how they are going to use data and users should be notified up front when a system will use it for any other purpose.
4. Use limitation: data should not be used for purposes other than those disclosed under the purpose specification principle without user consent.
5. Security safeguards: data should be protected with reasonable security safeguards (encryption, secure transmission channels, etc).
6. Openness: the user should be up front notified when the data collection and usage practices started away.
7. Individual participation: users should have the right to insert, update, and erase data in their profiles stored at data controllers.
8. Accountability: data controllers are responsible for complying with the principles mentioned above.

In this work, we present an enhanced middleware for collaborative privacy (*EMCP*) that allows creating reasonable referrals without breaching user privacy. *EMCP* employs a set of mechanisms to allow users to share their data among each other in the network to form a group to attain collaborative privacy. The users' cooperation is needed not only to protect their privacy but also to allow the service to run properly. Highly reputable peers aggregate participants' preferences then encrypt these collected profiles using homomorphic encryption in order to permit particular operations to be performed on encrypted data without need for prior decryption then they submit these profiles to PRS in order to produce referrals. The encrypted profiles hide the identities of participants, and thus hamper the ability for the untrusted PRS to invade users' privacy by profiling or tracking them. However, participants cannot trust each other as well and hence the aggregation process should not expose their preferences. Hence, we proposed a trust based obfuscation mechanism, which designed especially to obfuscate items' ratings before their submission to these highly reputable peers.

This approach preserves the aggregates in the dataset to maximize the usability of information in order to accurately predicate ratings for items that have not consumed before by the group members. In addition, *EMCP* employs interpersonal trust between users to enhance recommendation accuracy and preserve privacy. The enhancement in accuracy is achieved by employing trust based heuristics to propagate and spot users whom are trustworthy with respect to the target user. Moreover, trust based heuristics enhance privacy by transforming participants' data in different ways based on different trust levels to hide the raw ratings. Thus, In contrast to a single obfuscation level scenario where only one obfuscated copy is released for all users using fixed parameters for the obfuscation mechanism, now multiple differently obfuscated copies of the same data is released for different requests with different trust levels. The more trusted the target-user is the less obfuscated copy he can access. These different copies can be generated in various fashions. They can be jointly generated at different times upon receiving new request from target user, or on demand fashion. The latter case gives users maximum flexibility.

In rest of this work, we will generically refer to news programs, movies and video on demand contents as Items. This paper is organised as follows. In Section 2, related works are described. Section 3 presents the threat model assumed in this work .Section 4 introduces IPTV network scenario landing our private recommender service. The proposed solution based on *EMCP* is introduced in Section 5. In Section 6, the two-stage concealment process is described in details. Proof of security and correctness for the two-stage concealment process is demonstrated in Section 7. In Section 8, the Results from some experiments on the proposed mechanisms are reported. Finally, the conclusions and recommendations for future work are given in Section 9.

2 Related Works

The majority of the existing recommender services are based on collaborative filtering techniques that build users' profiles in two ways on ratings (explicit rating procedures) or on log archives (implicit rating procedures) [3]. These procedures lead to two different approaches for to collaborative filtering including the rating based approaches and log based approaches. The majority of the literature addresses the problem of privacy on collaborative filtering techniques, due to it being a potential source of leakage of private information shared by the users as shown in [4]. In [5] a theoretical framework is proposed to preserve the privacy of customers and the commercial interests of merchants. Their system is a hybrid recommender system that uses secure two party protocols and public key infrastructure to achieve the desired goals. In [6, 7] a privacy preserving approach is proposed based on peer to peer techniques using users' communities, where the community will have a aggregate user profile representing the group as a whole but not individual users. Personal information will be encrypted and communication will be between individual users but not servers. Thus, the recommendations will be generated on

the client side. In [8, 9] another method is suggested for privacy preserving on centralized recommender systems by adding uncertainty to the data by using a randomized perturbation technique while attempting to make sure that the necessary statistical aggregates such as the mean do not greatly get disturbed. Hence, the server has no knowledge about the true values of the individual items' ratings for each user. They demonstrate that this method does not essentially decrease the accuracy obtained in the results. But recent research work [10, 11] pointed out that these techniques do not provide levels of privacy as was previously thought. In [11] it is pointed out that arbitrary randomization is not safe because it is easy to breach the privacy protection it offers. They proposed random matrix based spectral filtering techniques to recover the original data from the perturbed data. Their experiments revealed that in many cases, random perturbation techniques preserve very little privacy. Storing users' profiles on their own side and running the recommender system in a distributed manner without relying on any server is another approach proposed in [12], where the authors proposed only transmitting similarity measures over the network and keeping users' profiles secret on their side to preserve privacy. Although this method eliminates the main source of threat against user's privacy, it requires higher cooperation among the users to generate useful recommendations. The work in [13] stated that existing similarities are deemed useless as traditional user profiles are sparse and insufficient. Recommender systems need new ways to calculate user similarities. They utilize interpersonal trustworthiness to describe the relationship between two users. The authors in [14] show the correlation between similarity and trust and how it can elevate movie recommendation accuracy.

3 Threat Model

In this work, we assume that an adversary aims to collect users' preferences in order to identify and track users. Thus, we consider our main adversary to be an untrusted PRS to which users send their preferences. We do not assume the PRS to be completely malicious. This is a realistic assumption because PRS needs to accomplish some business goals and increase its revenues. PRS can construct the profiles of the users based on the requests sent. Hence, the problem we are tackling has two sides; we want to detain the ability of the adversary to identify users based on a set of identifying interests and thus track them by correlating these data with data from other publicly-accessible databases and in the same time we want to prevent the adversary from profiling the users through their network identity and therefore invade their privacy. Intuitively, the system privacy is high if PRS is not able to reconstruct the real users' preferences based on the information available to it. Other adversaries are malicious users trying to collect preferences information about others. Malicious users can eavesdrop and collect preferences while being aggregated. So, while hiding our identity from the recommender service, it should not be revealed to other users sniffing the network.

4 Private Recommender Service for IPTV Network Scenario

We extend the scenario proposed in [15-20], where a private recommender service (PRS) is implemented as an external third party server and users give their preferences to that server in order to receive referrals. *EMCP* preserves users' privacy by utilizing three mechanisms: trust based obfuscation, aggregation and threshold encryption. The basic idea for a recommendation based on *EMCP* is that the user who needs recommendation will form a group with other participants in the IPTV network who decided to join his recommendation process. Then, the group members elect highly reputable peers (that we call super-peer) to aggregate their preferences they are willing to share into profiles. The super-peers will cooperate to achieve privacy by encrypting collected profiles using threshold homomorphic encryption in order to permit particular operations to be performed on encrypted data without need for prior decryption and then submit these aggregates to PRS in order to produce referrals. The encrypted profiles hide the identities of the participants, and thus hamper the ability for the untrusted PRS to profile and track users that invade their privacy. However, participants cannot trust each other as well and hence the aggregation process should not expose their preferences. Hence, we proposed a trust based obfuscation mechanism to obfuscate preferences prior submission to super-peers.

Our solution relies on a two stage concealment process, the first stage is trust based obfuscation and it takes place at participant side to hide extracted preferences prior their submission to super-peers. Then the second stage is threshold homomorphic encryption and it takes place at super-peers to hide collected profiles prior their submission to PRS. The overall process might be described as follows: upon receiving a request from the target user, a group of participants is formed that is managed by an elected super-peer. Super-peers negotiate with both the target user and PRS to express their privacy practices for the data collection and usage via P3P policies which are XML statements that answers questions concerning purpose of collection, the recipients of these profiles, and the retention policy. After receiving P3P policy& request, *EMCP* ensures that the extracted preferences for specific request do not violate the privacy of its host by checking whether there is an APPEL privacy preference corresponding to that given P3P policy, and then it starts collecting preferences that fulfil the request and in the same time satisfies the extracted APPEL preferences. The extracted items' ratings are obfuscated using a trust based obfuscation mechanism provided by *EMCP*, such that each item's rating is obfuscated based on the privacy preferences of its owner and estimated trust level with the target user. Furthermore, items identifiers and meta-data are hashed using locality-sensitive hashing. This step prevents the super-peers from knowing each participant's raw ratings for different items identifiers. The super-peer collects

these obfuscated preferences and computes an aggregation on them, which does not expose individual ratings. Next, the collected profiles are encapsulated using threshold encryption and submitted to PRS to predicate ratings for the referred items that did not consumed before and will be offered in the end to the target-user and participants. The collaborative filtering task at PRS will be reduced to computing addition on aggregated ratings without exposing the raw ratings. Therefore, our solution ensures privacy in the relation between the participants and PRS and in between the participants themselves. In the following section we will describe some enhancements attained using *EMCP*:

1. ***Usage of Pseudonymous for the Profiles:*** The real user's identity is not always required to provide referrals. Users can be identified by anonymous pseudonyms or nicknames, so that the binding of nickname and the real life identity is not always manifested.
2. ***User Private Data Store at the Client:*** Shifting from the approach of storing the user profiles in the server side to the one of storing the profiles on the clients' STBs helps reducing the privacy concerns. One key aspect is keeping the profiles encrypted to avoid people having access to the client's machine or malware that looks for user profiles.
3. ***Request-Oriented Collection:*** Upon receiving a request from the target user, query rewriter and preference checker assures that learning agent extracts only the required preferences from user's profile for a particular request the user is engaged in. The key point relies on knowing what kind of data is required for a given request that can contribute to improve the performance of the recommendation, because the recommender service does not provide recommendation based on one user's full profile information (e.g.: other users' preferences might not be relevant to the request). Likewise, once a user completes a particular request, he/she may no longer be interested in receiving recommendations related to that request for a period.
4. ***Communication through Anonymous Networks:*** internet records containing IPs, etc stored at service providers, contain information that permit the identification of user when submitting their obfuscated preferences to the node that requested recommendation. *EMCP* employ anonymous communication to hide the network identity for the participants by routing the submission of their obfuscated preferences through relaying nodes in an anonymous communication network before sending them to Super-peers. The main challenge for *EMCP* is to tune up and optimize the performance of the anonymous network while maintaining the user anonymity, we employed the path selection algorithm presented in [18] to enhances the anonymous network performance. Figure (1) shows the architecture of our approach.

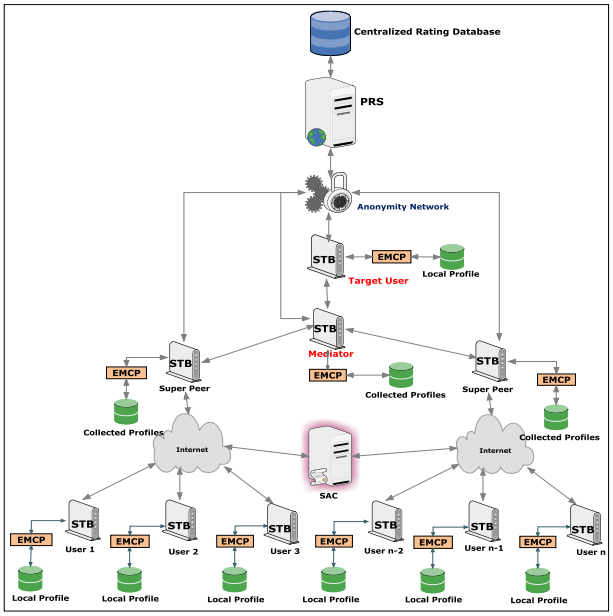


Figure 1: *EMCP* Middleware with Third Party Private Recommender Service.

Our solution relies on the hierarchical topology proposed in [21]; per each request participants are organized into peer-groups managed by super-peers. Electing super-peers is based on negotiation between the participants and security authority centre .The security authority centre (SAC) is a trusted third party responsible for making an assessment on those super-peers according to the participants' reports and periodically updating the reputation of each super-peer based upon it. Reputation mechanisms are employed to elect suitable super-peers based on estimating values for user-satisfaction, trust level, processing capabilities and available bandwidth, further details and information on complex reputation mechanisms can be found in [22]. When a problem occurs with a specific super-peer during the recommendation process, a participant can report it to SAC. After investigation, the assessment of the super-peer will be degraded. This will limit the chance for electing it as a super-peer in the future. On the other hand, successful recommendation processes will help upgrade the super-peer reputation. An IPTV provider can offer certain benefits (like free content, prizes,... etc) for those participants who have a sustained success rate as a super-peer.

Our solution depends upon the set top box (STB) device at the user side. STB is an electronic appliance that connects to both the network and the home television. With the advancement of data storage technology each STB is equipped with mass storage, e.g. Cisco STB. *EMCP* components are hosted on STB; Moreover STB storage stores the user profile. On the other hand, PRS maintains a centralized rating database that is used to provide referrals if the number of participants in group fall below a certain threshold. PRS is the third-party entity recruited by the IPTV network provider to operate referrals by consolidating the information received from multiple sources.

5 Proposed Solution

In the beginning, we want to introduce the notions of privacy and trust within our framework, we need to confirm what we mean by privacy and trust first. To define privacy and trust in our terms, we first approach the notion of privacy in following terms: "A participant who wants to join recommendation request in a network of users, does not has to reveal raw ratings in his/her profile during the recommendation process and elected super-peers does not wish PRS to learn any raw ratings in the collected profiles they provide". While in the context of this paper, trust is interpreted as "a user's expectation of another user's competency in providing ratings to reduce its uncertainty in predicating new items' ratings [23]". In our framework, the notion of privacy surrounding the disclosure of users' preferences and the protection of trust computation between different users are together the backbone of our solution. We apply a trust based obfuscation mechanism at participant side, which produces different copies of items' ratings based on the various trust levels with target user. The trust computation is done locally over the obfuscated participant's preferences, and then recommendation is served using secure multi-party computation protocol. Utilizing trust heuristic as input for both group formation and obfuscation process has been of great importance in mitigate some of malicious insider attacks such as infesting the trust computation results. As future work, we plan to investigate miscellaneous insider attacks and strengthen our framework against them.

In the next sub-sections, we will present our proposed middleware for protecting the privacy of users' preferences. Figure (2) illustrates the *EMCP* components running inside user's STB. *EMCP* consists of different co-operative agents. A Learning agent captures user interests about miscellaneous items explicitly or implicitly to build a rating database and meta-data database. The local obfuscation agent implements a trust based obfuscation mechanism to achieve user privacy while sharing his/her preferences with super-peers or PRS. The encryption agent is only invoked if the user is acting as a super-peer in the recommendation process; it executes *SR* protocol on the collected profiles. These mechanisms act as wrappers that conceal preferences before they are shared with any external entity. Since the database is dynamic in nature, the local obfuscation agent periodically desensitizes the updated preferences, and then a synchronize agent forwards them to the PRS upon owner permissions. Thus recommendation can be made on the most recent ratings. Moreover, synchronize agent is responsible for calculating & storing parameterized paths in anonymous network that attain high throughput[18], which in turn can be used in submitting preferences anonymously. The policy agent is an entity in *EMCP* that has the ability to encode privacy preferences and privacy policies as XML statements depending on the host role in the recommendation process. Hence, if the host role as a "super-peer", the policy agent will has the responsibility to encode data collection and data usage practices as P3P policies via XML statements which are answering questions concerning purpose of collection, the recipients of these profiles, and the retention policy. On the other hand, if the host role as a "participant" policy agent acquires the user's privacy preferences and express them using APPEL as a set of preferences rules which are then decomposed into set of elements that are stored in a database called "privacy preferences" as tables called "privacy meta-data". These rules contain both a privacy policy and an action to be taken for such privacy policy, in such way this will enable the preference checker to make self-acting decisions on objects that are encountered during data collection process regarding different P3P policies (e.g.: privacy preferences could include: Certain categories of items should be excluded from data before submission, Expiration of purchase history, Usage of items that have been purchased with the business credit card and not with the private one, Generalize certain terms or names in user's preferences according to defined taxonomy, Using synonyms for certain terms or names in user's preferences , suppressing certain items from the extracted preferences and insert dummy items that have same feature vector like the suppressed ones as described in [24], limiting the potentially output patterns from extracted preferences etc in order to prevent the disclosure of sensitive preferences in user's profile). Query Rewriter rewrites the received request constrained by privacy preference for its host.

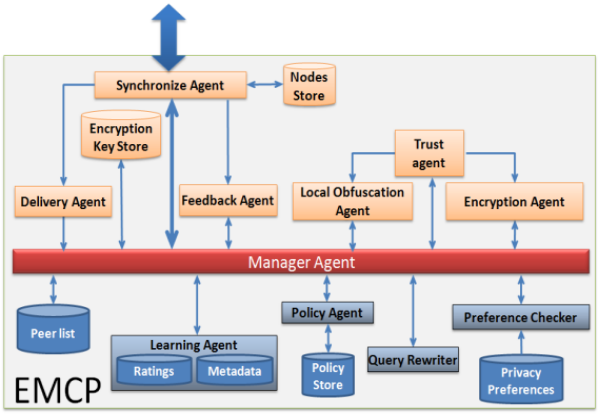


Figure 2: EMCP Components.

Figure (3) shows the participants interactions with super-peers and PRS. The recommendation process in our solution operates as follows:

1. The learning agent collects the user's interest about different items which represent his profile. The local profile is stored on two databases, the first one is the rating database that contains (item_id, rating) and the second is the meta-data database that contains the feature vector for each item [24] (item_id, feature1, feature2, feature3). The feature vector can include genres, directors, actors and so on. Both implicit and explicit ways for information collection [25] are used to construct these two databases and maintain them.
2. As stated in [16], the target user broadcasts a message to other users in the IPTV network requesting recommendation for a specific genre or category of items. Thereafter, the target user selects a set of his

preferences to be used later in the computation of trust level at the participant side. So as to hide the items identifiers and meta-data from other participants, The local obfuscation agent uses locality-sensitive hashing (LSH) [26] to hash these values. One interesting property for LSH is that similar items will be hashed to the same value with high probability. Super-peers and PRS are still able to perform computation on the hashed values using appropriate distance metrics like hamming distance or dice coefficient. Simultaneously, local obfuscation agent sanitizes items' ratings using trust based obfuscation. Finally, the target user dispatches these obfuscated items' ratings along with their associated hashed values to the Individual users who have decided to participate in the recommendation process.

3. Each group of participants negotiates with SAC to select a peer with the highest reputation as a "super-peer" which will act as a communication gateway between the PRS and the participants in its underlying group.
4. Each super-peer negotiates with both the target user and PRS to express its privacy policies for the data collection and usage process via P3P policies which are XML statements that answers questions concerning purpose of collection, the recipients of these profiles, and the retention policy. Thereafter, super-peers engage in key distribution phase of the *SR* protocol, at the end of this phase each super-peer will possess a share of the decryption key along with the complete encryption key to encrypt the collected profiles. The encrypted profiles can only be decrypted only if any subset consisting of a threshold t of super-peers cooperate.
5. At the participant side, the manager agent receives the request from the target user along with the P3P policy form the elected super-peer; then it forwards P3P policy to preference checker and the request to query rewriter. The preference checker ensures that the extracted preferences for a specific request do not violate the privacy of its host by checking whether there is an APPEL preference corresponding to the given P3P policy and sends it to the query rewriter. The user's preferences can be transferred or collected only if the purpose of statement for the collectors satisfies the privacy preferences. The query rewriter will have knowledge about privacy preferences related to current request via APPEL preference then it rewrites the received request constrained by the privacy preference for its host in order to only retrieve the preferences that the host agrees to share as well as prevent the disclosure of confidential preferences in the participant's profile. This step enable the participant to decide when the recommendation takes place, which information should be collected and for which purpose. This step will ensure the privacy principles compliance and put the user in control the information that is part of their profiles. The modified request is directed to the learning agent to start collecting preferences that could satisfy the modified query. The manager agent ensures that the collected preferences compliance with the collection data principle, as only the required preferences for the particular request the user is engaged in, is extracted for the local obfuscation process.
6. In the meanwhile, the trust agent calculates approximated interpersonal trust between its host and the target user based on the received preference. It is done in a decentralized fashion using the entropy definition proposed in [23] at each participant side. The entropy value becomes lower as the users' ratings are more consistent, which is similar to the definition of trust previously stated. $\forall_{j=1}^{n} T\left(u_a, u_{b_j}\right)$ is the estimated trust between the target user u_a and participant u_{b_j}. the whole process can be described using the following steps:
   i. Each participant $\forall_{j=1}^{n} u_{b_j}$ determines a subset of his/her items' ratings that will be required for recommendation process. Then the participant utilizes shared items rated by both of u_a, u_{b_j} for the trust computation. Determining shared rated items is done by matching the received items' hash values from target user u_a with his/her local items' hash values.
   ii. Participant u_{b_j}computes the trust level using equation $T\left(u_a, u_{b_j}\right) =$

$$\frac{Entropy(u_a) - Entropy\left(u_a \middle| u_{b_j}\right)}{Entropy(u_a)} = \frac{\left(1 - \frac{log\, N}{log\, ZN}\right) + \frac{1}{N\, log\, ZN}\left(\sum_{i=1}^{Z}\sum_{j=1}^{Z} n_{ij}\, log\, n_{ij} - \sum_{i=1}^{Z} n_i\, log\, n_i\right)}{1 - \frac{1}{N\, log\, ZN}\sum_{i=1}^{Z} n_i\, log\, n_i} \quad (1)$$

   Equation (1) is an adapted formalization of trust as proposed in [23] where *Z* denotes the number of states of rated values and *N* is the total number of rating times. For example if *Z=6* and *N=20* when 20 ratings are made with 1 to 6 integer valued scores. Employing entropy to select trustworthy neighbours achieves an improvement in the group formation and rating predication. The enhancement in rating predication is stemmed from trust propagation, so if $u_{b_j=x}$ is selected as a trustworthy user and he/she does not have a rating for the item to be predicted, a trustworthy user $u_{b_j=y}$ of user $u_{b_j=x}$ can also be used for the predication.
   iii. Each participant $\forall_{j=1}^{n} u_{b_j}$ sends his/her calculated trust value to the super-peer. The Estimated trust values are forwarded to both the super-peers and PRS.
   iv. Each participant $\forall_{j=1}^{n} u_{b_j}$ sends this trust value to the local obfuscation agent to adjust the obfuscation level with trust level, in other words, we correlate the obfuscation level with different levels of trust, so the more trusted a target user is, the less obfuscated copy of users' preference he can access. The local obfuscation agent executes enhanced value-substitution (*EVS*) algorithm on items' ratings that are required in the recommendation process. Moreover the local

obfuscation agent hashes their identifiers and meta-data using LSH. The level of obfuscation is determined using the trust level with the target user, and then participants submit their obfuscated preferences to the super-peers of their group. Anonymous communication [18] utilized to hide the network identities of group members when submitting their obfuscated preferences to the super-peers.

v. Finally, the policy agent audits the original and modified requests plus estimated trust level and P3P policy with previous requests; this step allows *EMCP* to prevent multiple requests that might extract sensitive preferences. In such a case, if the target user requests same data twice, its trust level will be reduced, in which will increase the level of the obfuscation in the extracted preferences. This step will cause extracted preferences appear as a completely different set of preferences to the target user.

7. Upon receiving the obfuscated preferences from the participants, each super-peer filters the received preferences based on the trust level of their owners such that $T\left(u_a, u_{b_j}\right) > \theta$ where θ is a minimum trust threshold value defined by the target user or PRS. Then, each super-peer collects the participants' pseudonyms and builds a group rating profile such that all the <hashed value, rating> elements belonging to similar items are grouped together. This allows the computing of the items popularity curve at each super-peer. The super-peer can seamlessly interact with the PRS by posing as an end-user and has a group profile as his own profile. Each super-peer $\forall_{x=1}^{k}\, SP_x$ calculates the following intermediate values for each user in the *N*-neighbourhood of target user $\forall_{j=1}^{n} u_{b_j} \in \text{Neighbor}(u_a)$,

$$\text{Then } \forall\, q = 1 \ldots T \quad \widetilde{r_{u_{b_j},q}} = r_{u_{b_j},q} - \bar{r}_q$$

$$\widehat{r_{q,u_{b_j}}}^{x} = \frac{T\left(u_a,u_{b_j}\right) * \widetilde{r_{u_{b_j},q}}}{T\left(u_a,u_{b_j}\right)} \qquad (2)$$

Where $r_{u_{b_j},q}$ is the rating value of participant u_{b_j} for item q. $\bar{r}_q$ is the average rating for item q in each items' cluster. Next, each super-peer encrypts theses intermediate ratings $\widehat{r_{q,u_{b_j}}}^{x}$ using the encryption key pk. Finally, the super-peer submits these ratings along with their associated hashed values to PRS, which in turn collects them to produce final referrals.

8. Upon receiving the encrypted ratings $\forall_{x=1}^{k}\forall_{j=1}^{n}\varepsilon_{pk}\left(\widehat{r_{q,u_{b_j}}}^{x}\right)$ from all super-peers, PRS stores them along with their participants' pseudonyms and hashed values in the centralized rating database. The rating predication phase is performed using the additive homomorphic property of the threshold paillier encryption as the required computations are additive. Thus, PRS executes an additive operation on the encrypted rating profiles without decrypting them so the private data of multiple super-peers can be preserved during the computation. Calculating the predicted rating for referrals done as shown in equation (3):

$$p_{u_a,q} = \varepsilon_{pk}\left(\overline{r_{u_a}}\right) * \left(\prod\nolimits_{j=1}^{n} \varepsilon_{pk}\left(\widehat{r_{q,u_{b_j}}}^{x}\right)\right)$$
$$= \varepsilon_{pk}\left(\overline{r_{u_a}} + \left(\sum\nolimits_{j=1}^{n} \widehat{r_{q,u_{b_j}}}^{x}\right)\right) \qquad (3)$$

Notice that the result will be equal to the weighted sum of the participants' rating plus the average rating of the target user r_{u_a}.Super-peers uses the reblinding property of the paillier encryption to prevent PRS and target user from obtaining any knowledge of $\widehat{r_{q,u_{b_j}}}^{x}$ values by trying a few possible values.

9. PRS forwards the encrypted referrals list along with their predicated ratings to super-peers which in turn perform threshold decryption on these results. The threshold decryption process requires that at least t of the super-peers are honest. Only when the required number of super-peers cooperates, they can perform decryption using their local share of the private key, and then they will be able to have the final referrals list for the entire group. Super-peers publish the final list to the target user and participants. Finally, each participant report scores about the elected super-peer of his group and target-user to SAC, which helps to determine reputation of each entity involved in referrals generation.

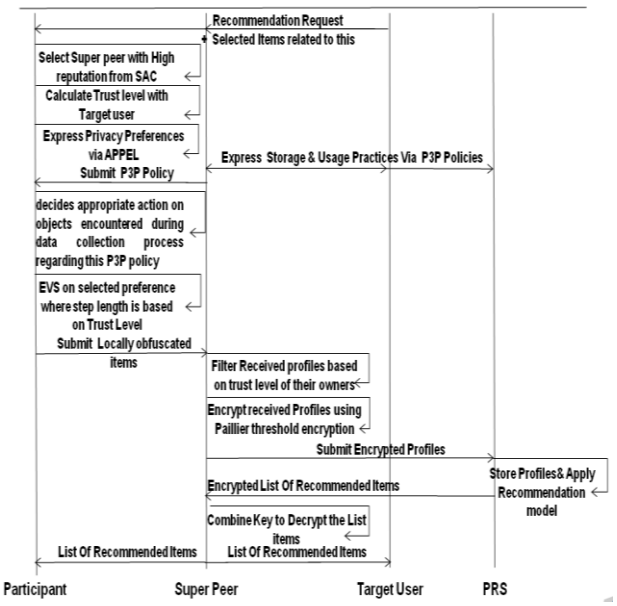


Figure 3: Interaction Sequence Diagram.

6 Proposed Two Stage Concealment Process

In the next subsections, we present our two stage concealment process used in *EMCP* to disguise the user items' ratings in way that secure the user's preferences in the untrusted PRS with minimum loss of accuracy. In our framework, each user has two datasets representing his/her profile. A local profile: represents the actual ratings of the user for different items; it is stored on his STB. Each user disguises this local profile before sending it to super-peer. An encrypted centralized

profile: this is the output of the two-stage concealment process that stored at PRS, the user gets recommendation directly from the PRS based on the previously collected profiles. We perform experiments on real datasets to illustrate the applicability of our mechanisms and the privacy and accuracy levels achieved by using them.

6.1 Cryptography Tools

Using additively homomorphic cryptosystem permit the computation of linear combinations of encrypted data without need for prior decryption, such that PRS can combine received encrypted rating profiles into a new ciphertext that is the encryption of the sum of the ratings of the original ratings. Formally, an encryption schema $\varepsilon_{pk}(.)$ denotes the encryption function with encryption key pk and $D_{sk}(.)$ denotes the decryption function with decryption key sk. Additive homomorphic cryptosystem possesses the following properties:

1. Given the encryption of plaintexts m_1 and m_2, $\varepsilon_{pk}(m_1)$ and $\varepsilon_{pk}(m_2)$. The sum $m_1 + m_2$ can be directly computed as $\varepsilon_{pk}(m_1 + m_2) = \varepsilon_{pk}(m_1) * \varepsilon_{pk}(m_2)$.
2. Given a constant k and the encryption of m_1, $\varepsilon_{pk}(m_1)$. The multiplication of k with the plaintext m_1 can be directly computed as $\varepsilon_{pk}(k.m_1) = \varepsilon_{pk}(m_1)^k$.

Paillier [27] proposed a probabilistic asymmetric algorithm for public key cryptography that is an example of an efficient additively homomorphic cryptosystem, this scheme is further extended by [28] with a threshold versions, but required the use of a trusted dealer to distribute the keys to the participants. The reliance on a trusted dealer was lifted in [29] to ensure that no single party or coalition of less than specific participants can recover the encrypted values. In designing *SR* protocol, we require a fully distributed key generation protocol. In particular, the coalition between PRS or target user with any super-peer within the group should not be able to decrypt the whole collected profiles submitted to PRS, but it only reveals the obfuscated profiles collected by this super-peer. Therefore neither can be used as a trusted "dealer" for key generation. Thus, we employ a fully distributed threshold cryptosystem, Since it is desirable to distribute trust between numerous super-peers and no single super-peer is assumed to be fully trusted, then the decryption key sk is shared among a number P of super-peers, and encrypted profiles can only be decrypted only if any subset consisting of a threshold t of super-peers cooperate but no subset smaller than t can perform decryption. Moreover, with the additively homomorphic property of Paillier schema it permits *SR* protocol to perform secure aggregation and predication over encrypted rating profiles. We assume a semi-honest model for the super-peers. Hence, we do not require zero-knowledge proofs (ZKPs) for the various cryptographic operations from the participants. We will briefly present the distributed paillier threshold cryptosystem below.

Key Generation

In this step, each super-peer $\forall_{i=1}^{n} SP_i$ generates n additive shares of two $\kappa/2$-bit strong primes, such that each super-peer have share p_i and q_i. Then use the method proposed in [29] to compute $N = pq$, $\lambda = lCM\ (p-1, q-1)$, $g = N + 1$ such that $p = \sum_{i=1}^{n} p_i$, $q = \sum_{i=1}^{n} q_i$, also d such that $d \equiv 1 \bmod N$ and $d \equiv 0 \bmod \lambda$. The public key $pk = (N, g)$ and the private key $sk = d$. Note that, super-peers perform biprimality test in [30] for checking if N is a product of two primes in a distributed way. If the test fails, the protocol is restarted

Key Sharing

The private key sk is shared among n super-peers with the Shamir scheme as $t-1$ degree polynomial where each party obtain (t, n) share of d : Let $a_0 = d$, and randomly choose a_i in $\{0, \ldots N-1\}$ and set $f(X) = \sum_{i=0}^{t} a_i X^i$. The share s_i of the ith super-peer SP_i is f(i) mod N.

Encryption

To encrypt a message $M \in Z_N$ with public key , randomly choose $r \in Z_N^*$ and compute $C = g^M r^n \bmod N^2$.

Share Decryption

To decrypt C, each super-peer SP_i computes the decryption share $c_i = c^{2\Delta s i} mod\ N^2$, where $\Delta = t!$ using his/her secret share s_i. And finally, if $t + 1$ valid shares are available, they can be combined to recover M as described in End decryption.

End Decryption

Let S be a set of $t + 1$ valid shares. Compute

$$M = L\left(\prod_{i \in S} c_i^{2\lambda_i} mod\ N^2\right) \frac{1}{4\Delta^2} mod\ N$$

Where $\lambda_i = \Delta \prod_{i' \in S \setminus i} \frac{-i}{i-i'}$, See [29] for more details on the correctness of the scheme and for proofs of security.

6.2 Local Obfuscation using Enhanced Value-Substitution (*EVS*) Algorithm

We propose a novel algorithm for obfuscating the users' ratings before sending them to the super-peers. This algorithm is called *EVS*, which has been designed especially for the sparse data problem we have here. Moreover the algorithm tunes its obfuscation parameters based on trust level. The available anonymisation algorithms perform single obfuscation levels for all participants and release one obfuscated copy for all of them which result in increasing data distortion and construction of inaccurate recommendation model. The key idea for *EVS* is based on the work in [31] that uses Hilbert curve as a dimensionality reduction tool to create a cloaking regions to attain privacy for users. Hilbert curve also has the ability to maintain the association between different dimensions. In this subsection, we extend this idea as following, we also use Hilbert curve to map m-dimensional profile to 1-dimensional profile then *EVS* discovers the distribution of that1-dimensional profile. Finally, we perform perturbation based on that distribution in such a way to preserve the profile range that led to providing accurate results when performing

rating predication. The output of our obfuscation algorithm should satisfy two requirements:

- Reconstructing the original profile from the obfuscated profile should be difficult, in order to preserve privacy.
- Preserving the distances of the data to achieve accurate results for the recommendation.

The steps for *EVS* algorithm consists of the following:

1. We denote the collected m-dimensional user preferences as dataset D of c rows, where each row is a sequence of m dimensions $A = A_1, A_2, A_3, A_4 \dots \dots, A_m$.
2. Trust level values are divided to a number of intervals defined by the user, associated with each interval an order k value. *EVS* chooses a value for order k according to the trust level associated with the target user.
3. *EVS* divides the m-dimensional dataset D into grids of order k as shown in [31, 32]. For order k, the range for each dimension divided into 2^k intervals.
4. For each dimension $\forall_{i=1}^{m} A_i$ of the collected preferences D:
   - Compute the k-order Hilbert value for each data point $\forall_{x=1}^{c} a_{ix}$. This value represents the index of the corresponding interval where it falls in.
   - *EVS* sort the Hilbert values from smallest to biggest, then use the step length (a user defined parameter) to measure whether any two values are near from each other or not. If these values are near, they are placed in the same partition $\forall_{v=1}^{k} k_{iv}$.

   These two steps iterates for all m-dimensions. The final result from these steps is k partitions for each dimension denoted as $\forall_{i=1}^{m} \forall_{v=1}^{k} C_{iv}$
5. *EVS* constructs a N shared nearest neighbour sets S_r where $r = 1 \dots . N$ as in [33] from different partitions with a new modified similarity function as following, two partitions in different dimensions C_{iv}, C_{i+1v} form a shared nearest neighbour set S_r if they share k-number of common elements such that $S_r = C_{iv} \cup C_{i+1v}$
6. For each newly created set S_r , *EVS* calculates its interquartile range. Then, for each point $a_i \in S_r$ generate a uniform distributed random point n in that range that can substitutes a_i.
7. Finally, the new set $D' = \cup_{r=1}^{N} S_r$ is sent to Super-peer.

6.3 Secure Recommendation Protocol (*SR*)

We proposed a protocol that enables PRS to calculate predicted ratings from the encrypted rating profiles. We called this protocol secure recommendation protocol (*SR*). *SR* protocol starts with the selection of super-peers using SAC as it is heavily relies on the underlying network topology; also it requires a set of super-peers to aggregate all participants' preferences at the bottom of the hierarchy into profiles in order to remove any possibility of a single super-peer being the bottleneck. To achieve reasonable efficiency, super-peers reserve the ability to independently reweight items' ratings based on trust values and omit the ones with low trust values, where such centralized computation can make the most significant difference. Moreover, they compute aggregated items' ratings from the obfuscated ratings received from their participants. Thereafter, super-peers engage in distributed key generation process using distributed threshold cryptosystem to generate public key to encrypt these profiles before submitting them to PRS. This key generation process will leave each super-peer with a share of the private key along with the complete public key. This makes sure that no single super-peer to able decrypt the profiles taken from different super-peers or the final referrals list retrieved from PRS. After all the super-peers collect preferences from participants and compute the aggregated ratings profiles, they engage independently in encrypting these results. Then, each super-peer will forward the ciphertext corresponding to the ratings profile over the entire group to the PRS. The PRS starts the rating predication phase on the ciphertext then forward back the results to the super-peers. The super-peers will then perform threshold decryption of these results. Only when the required number of super-peers cooperates, they can perform decryption using their local share of the private key, and then they will be able to have the final referrals list for the entire group. Note that we have focused on the decryption process to make sure that no single super-peer can get the profiles over a subset of super-peers in the group and malicious super-peers in the network are unable to compromise the security of the protocol. Moreover, utilizing fully distributed threshold cryptosystem ensures that all collected profiles become useless after the termination of recommendation process even if an attacker obtains the collected profiles. *EMCP* automatically destroys key shares directly after decrypting the received referrals list, without any explicit action by the participants or any party storing or archiving that data

Protocol _SecureRecommendation

Do forever

/* Applied in cases where super-peers are not already defined, Electing super-peers is based on negotiation between participants and SAC to select peers with the highest reputations*/

SuperPeer = selectSP ();

/* Find out who are other super-peer from SAC */

SPList = find SuperPeer ();

/* Check if I am super-peers to start collecting participants' preferences & generate keys for encryption agent */

If (me == SuperPeer)

/* Delivery agent listens to receiver channel to collect obfuscated preferences from participants associated with this super-peer */

ListenToReceiverChannel (CollectChannel,

ReceivedObfuscatedPreferences $r_{u_{b_j},q}$, $T\left(u_a, u_{b_j}\right)$);

/* Delivery agent combine the obfuscated preferences on the receiver channel if trust level for its participant higher than specific threshold value θ set by the target user */

If T $\left(u_a, u_{b_j}\right) > \theta$ then store $r_{u_{b_j},q}$

/* Delivery agent combine the obfuscated preferences on the receiver channel, if there is a change in the local preferences or if there is a new preferences received */

if (LocalObfuscatedPreferences == true || NewReceivedObfuscatedPreferences == true)

/* Calculates the normalized rating for item q from rating of each participant u_b*/

$\forall_{j=1}^{n} u_{b_j} \in Neighbor(u_a)$ calculate $\widetilde{r_{u_{b_j},q}} = r_{u_{b_j},q} - \bar{r_q}$

/* Combine the Received ratings with previously collected ratings for each item q*/

$\forall\, q = 1 \ldots T$

CombinedPreferences $\widehat{r_{q,u_{b_j}}}$ = CombinePreferences

$$\left(\text{LocalObfuscatedPreferences} \left(\frac{T\left(u_a, u_{b_j}\right) * \left(\widetilde{r_{u_{b_j},q}}\right)}{T\left(u_a, u_{b_j}\right)} \right), \text{ReceivedObfuscatedPreferences } \widetilde{r_{u_{b_j},q}} \right)$$

End if

End if

/* Generate public/private Key pair using a distributed protocol employing all other super-peers. The function SPDKG () leaves every super-peer with the entire public key and a share of the private key */

(PublicKey, PrivateKey) = SPDKG(SPList);

/* Initiate the encryption agent to encrypt my combined preferences with the public key and submit it to PRS */

Submit $\widehat{r_{q,u_{b_j}}}'$ (Enc(PublicKey, CombinedPreferences $\widehat{r_{q,u_{b_j}}}$))

To PRS;

/* PRS receive collected preferences for different super-peers */

PRS receives $\widehat{r_{1,u_{b_1}{}^1}}'$, $\widehat{r_{1,u_{b_1}{}^2}}'$, … , $\widehat{r_{2u_{b_1}{}^1}}'$, $\widehat{r_{2u_{b_1}{}^2}}'$, … , $\widehat{r_{Tu_{b_j}{}^k}}'$

such that $\widehat{r_{qu_{b_j}{}^k}}'$ is the encrypted rating for item q ∈ {1, … … T}

by user u_{b_j} $\left(\forall_{j=1}^{n} u_{b_j} \mid T\left(u_a, u_{b_j}\right) > \theta\right)$ from super-peer

SP_x $\left(\forall_{x=1}^{k} SP_x\right)$

/* PRS Calculates Predicated ratings for each unrated Item in the collected profiles*/

For each item q = 1 to T do

PRS Calculates $\forall_{x=1}^{k}\, p_{u_a,q} = \left(\text{Enc}(\text{PublicKey}, \widehat{r_{u_a}}\right) * \left(\prod_{j=1}^{n} \text{Enc}(\text{PublicKey}, \text{CombinedPreferences } \widehat{r_{q,u_{b_j}{}^x}})\right) \forall_{j=1}^{n} u_{b_j}$

End for

/* Upon receiving the list of predicated ratings for referred items, Target user request super-peers to start decrypt the entire list */

if (me == SuperPeer) Reclist = thresholdDecrypt(encryptedratingslist $\left(p_{u_a,q}\right)$, SPList)

End Protocol_SecureRecommendation

Algorithm selectSP

/* Each participant contact SAC to obtain list of peers of highest reputation to be elected as super-peer for the group */

Requst(HR_Peerlist);

/* Each super-peer broadcast to the neighbors indicating its existence as their neighbor*/

broadcast(SP_id);

/* if participant receives more than super peer id it compare P3P policies for each adjacent super-peer & select the one with suitable P3P policy to his privacy preferences */

listenToReceiverChannel(defaultChannel, SP_id);

if (ReceiverPeerId(SP_id) ≠ 1) Compare(SP_CollectionPolicies);

PeerGroupJoinRequest(SP_id);

End if

/*Each super-peer Listens to the receiver channel to form a group */

listenToReceiverChannel(defaultChannel, numNeighbors);

broadcast(numNeighbors);

listenToReceiverChannel(defaultChannel,PeerGroupCountPair[ ]);

superpeer= PeerGroupCountPair[maxIndex].getNeighborID();

return selectSP;

7 Proof of Security and Correctness

The proof of security for both *EVS* algorithm and *SR* protocol depends on how much information is leaked during the execution of the prediction phase. At the same time, our proposed mechanisms should output accurate results.

7.1 Privacy Breach Evaluation for *EVS* Algorithm

Privacy breach can be described in terms of how well the original user's ratings can be estimated from the submitted obfuscated ratings. Unlike other techniques, our method generates new data points, whose interpoint distances approximate the original distances. Consequently, points which lie close to one another in the original space mostly remain close to each other in the transformed space. Therefore, it seems theoretically to be more resilient to some potential attacks [34] that exploit the properties of the released data. These attacks are based on how much information about original data is available to the attacker that is obtained through either known input-output and known sample. In the known input-output, attacker knows collection of linearly dependant original data points and points they map in perturbed data. While in known sample, assumes that original data arose as independent samples of multidimensional random vector with unknown probability density function, and the attacker has access to a collection of these independent samples. In *EVS* algorithm, the linear ordering based on Hilbert curve retains the proximity and neighboring aspects of the original data. We define H_d^N for $N \geq 1$ and $d \geq 2$ as the N^{th}order Hilbert curve (defined values based on trust level) for a d- dimensional space. $H_d^N: [0, 2^{Nd} - 1] \rightarrow [0, 2^N - 1]^d$ as follows : Hilbert value $H = \epsilon(P)$ for $H \in [0, 2^{Nd} - 1]$, where P is coordinate of each point in $[0, 2^N - 1]^d$. Thereafter, we cluster nearby Hilbert values based on step length (a user-defined parameter) then *EVS* substitutes each point in the group with uniform distributed random point in the same

interquartile range for that cluster. Therefore we can consider ϵ as a one-way function if the curve parameters are unknown. These parameters include (starting point, N, step length) are defined at the participant side and any external entity only know the final perturbed data that participant agree to release. As a result, the statistical information from the perturbed data are inconsistent with that from the original data. Therefore, attacks such as those described before would be in efficient in breaching privacy. In addition to that, clustering Hilbert values and substituting each point with random point introduces uncertainty about exact distance between data points, thus will make any distance based attach ineffective.

7.2 Proof of Security & Correctness for *SR* Algorithm

Theorem 1: additive operation performed by PRS in *SR* protocol is correct and accurate without the need of decryption keys.

Proof: based on the first property of additive homomorphic cryptosystem, we can determine that additive operations for encrypted data are correct as follows: given the encryptions $\varepsilon_{pk1}(m_1) = a$ and $\varepsilon_{pk1}(m_2) = b$ where $\forall m_1, m_2 \in \mathbb{Z}_n$, given encryption key $pk2$

$\varepsilon_{pk2}\left(\varepsilon_{pk1}(m_1)\right).\ \varepsilon_{pk2}\left(\varepsilon_{pk1}(m_2)\right)\ mod\ N^2 =$
$\varepsilon_{pk2}(a).\ \varepsilon_{pk2}(b) = (g^a r_1^N).(g^b r_2^N)\ mod\ N^2 =$
$g^{a+b}(r_1 r_2)^N\ mod\ N^2 =$
$\varepsilon_{pk2}\left(\varepsilon_{pk1}(m_1 + m_2)\right)\ mod\ N^2$

Based on that, the PRS does not require any decryption key in order to aggregate all encrypted data.

Theorem 2: *SR* protocol computes predicated ratings for each referred item based on similar users' ratings without revealing extra information to any party.

Proof: Since each participant obfuscates his items' ratings and hashes their meta-data before submitting them to the super-peers. Moreover, each super-peer encrypts the collected profiles with the common encryption key and computation is performed on encrypted data and the decryption key is distributed between different super-peers. This makes sure that no single party will be able to decrypt these encrypted profiles taken from different super-peers or the final referrals list retrieved from PRS. This particular property is possible because of the threshold nature of the employed cryptosystem. In our two stage concealment process, the super-peer aggregates all obfuscated preferences then performs intermediate-computations on the obfuscated ratings for each item without having to know their real ratings or identifiers. No party can see extra information during the execution of the *SR* protocol. As for participants, they participate in the recommendation process without knowing other participants' identity. Since not all the participants have the same super-peer nor do they have direct communication with each other. The local profile is secured and can only be viewed by its owner before applying the trust based obfuscation mechanism. In addition, employing reputation techniques to select super-peers with a high success rate in previous recommendation processes ensures the selection of reliable peers that will perform the required phases. PRS cannot see the received profiles as the decryption key is unknown. Furthermore, the decryption process requires a subset consisting of a threshold t of super-peers to cooperate. After PRS generates the final referrals list, PRS submits it to super-peers which in turn perform threshold decryption process. Then they publish this list to all participants.

Theorem 3: Assuming that all parties follow the protocol, *SR* protocol can correctly compute the predicated rating for each referred item.

Proof: When each super-peer encrypts collected rating profiles with encryption key $\varepsilon_{pK}\left(\widehat{r_{q,u_{b_j}}^{\ x}}\right)$. PRS performs the additive operation on encrypted rating profiles based on paillier's homomorphic cryptosystem as follows:

$$p_{u_a,q} = \varepsilon_{pk}\left(\overline{r_{u_a}}\right) + \left(\varepsilon_{pk}\left(\widehat{r_{q,1}}\right) + \varepsilon_{pk}\left(\widehat{r_{q,2}}\right) + \varepsilon_{pk}\left(\widehat{r_{q,3}}\right) + \cdots + \varepsilon_{pk}\left(\widehat{r_{q,u_{b_j}}}\right)\right)$$

$$p_{u_a,q} = \varepsilon_{pk}\left(\overline{r_{u_a}} + \left(\sum_{j=1}^{n} \widehat{r_{q,u_{b_j}}^{\ x}}\right)\right)$$

$$p_{u_a,q} = \varepsilon_{pk}\left(\overline{r_{u_a}}\right)\left(\prod_{j=1}^{n} \varepsilon_{pk}\left(\widehat{r_{q,u_{b_j}}^{\ x}}\right)\right)$$

After the threshold decryption by super-peers, we will obtain the final predicated rating as in equation

$$p_{u_a,q} = \overline{r_{u_a}} * \left(\prod_{j=1}^{n} \widehat{r_{q,u_{b_j}}^{\ x}}\right)$$

So the result from *SR* protocol is correct.

8 Experiments

In this section, we describe the implementation of our proposed solution. The experiments are run on 2 Intel® machines connected on local network, the lead peer is Intel® Core i7 2.2 GHz with 8 GB Ram and the other is Intel® Core 2 Duo™ 2.4 GHz with 2 GB Ram. We used MySQL as data storage for the users' preferences that acquired by learning agent. The proposed two stage concealment process is implemented in C++ using the MPICH implementation of the MPI communication standard for distributed memory implementation of the *SR* protocol to mimic a distributed reliable network of peers. To implement Paillier encryption scheme, the Number Theory Library (NTL) was used. One practical issue that must be dealt with when using the Paillier cryptosystem is the fact that it cannot naturally encrypt floating-point numbers. Floating-point numbers must be converted to a fixed-point representation. This is done by multiplying them by a large constant C and then truncating the result to an integer. In these experiments, C = 100000. Other methods in [35] can also be used. The experiments presented here were conducted using the Jester dataset provided by Goldberg from UC Berkley [36]. The dataset contains 4.1 million ratings on jokes using a real value between (-10 and +10) of 100 jokes from 73.412 users. The data in our experiments consists

of ratings for 36 or more items by 23.500 users. We evaluated the proposed solution from different aspects: privacy achieved, accuracy of results and performance. We used the mean absolute error (MAE) metric proposed in [37]. MAE is one of most famous metrics for recommendation quality. As it measures the predication verity between the predicated ratings and the real ratings, so smaller MAE means better recommendation provided by PRS. To measure the privacy or distortion level achieved using our mechanism, we used the variation of information metric VI [38] to estimate data error. A higher value of VI means a larger distortion between the obfuscated and original dataset, which means a higher level of privacy. The experiments involve dividing the data set into a training set and testing set. The training set is obfuscated then used as a database for PRS. Each rating record in the testing set is divided into rated items t_i and unrated items r_i.The set t is presented to PRS for making predication p_i for the unrated items r_i. For the representation process of the trust calculation, we add the default value 0 for items not rated. In our dataset, the first column of every raw stores how many items are rated by the user, which is necessary for the trust estimation process. We divided trust levels into three intervals [highest, moderate, and lowest] and associated hilbert curve order for each interval. The experiments were performed while keep the number of super-peers $n = 9$, as described earlier they will be responsible for aggregating the data of 23.496 participants. We assume the trust level for all participants to be above the minimum trust threshold θ, which is required for the inclusion in the prediction process. The recommendation process can be initiated by any user that will act as the target-user for the referrals list. The trust level between participants and target-user is calculated locally on their STB devices.

In the first experiment, we want to measure the elapsed time for distributed key generation by varying the encryption key length and number of participants. Therefore we run the function SPDKG in *SR* protocol on 9 super-peers with different key length, and then we measure the elapsed time for distributed key generation and plot results in figure (4). Moreover, we set the key length to 1024 bits with varying number of super-peers (3 to 17) and we plot the elapsed time results in figure (5).

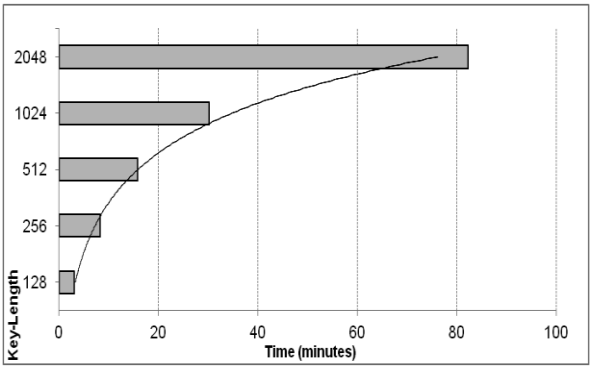


Figure 4: Key generation time for different Key Length.

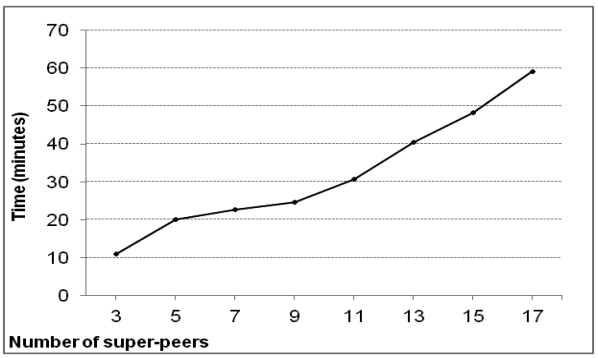


Figure 5: Key generation time for various numbers of super-peers.

In the second experiment, we want to measure the elapsed time for calculating the predicated ratings in *SR* protocol by varying the encryption key length and number of participants. We run the predication phase several times by encrypting all 12.674 records with different key length and distribute them equally on 9 super-peers, then PRS start collecting these records to perform predication phase. The results for elapsed time are shown in figure (6). Moreover, we set the key length to 1024 bits with varying number of super-peers (3 to 17) and we plot the elapsed time results in figure (7).

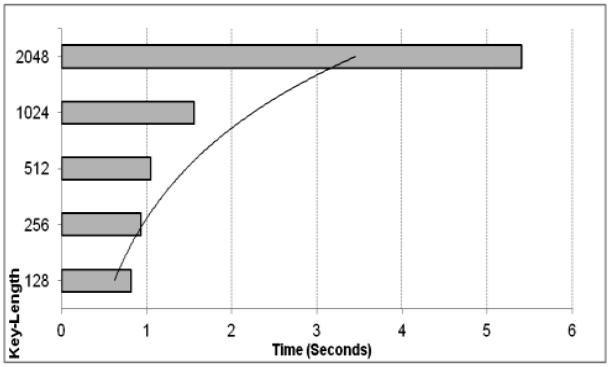


Figure 6: Ratings predication time for different Key Length.

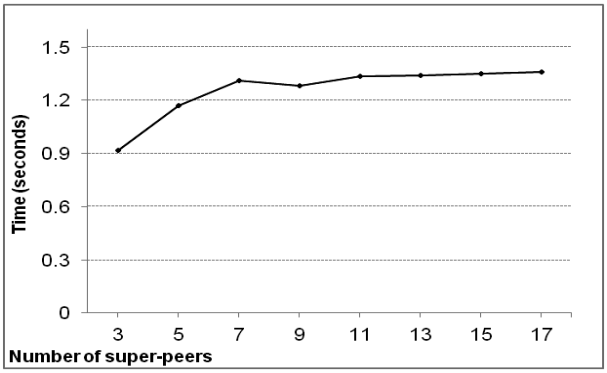


Figure 7: Ratings predication time for various numbers of super-peers.

In the third experiment, we aim to analyze execution time for *SR* protocol for varying set of data sizes. Therefore, we vary the minimum trust threshold to obtain a different number of participants' records in the recommendation process, then we run the *SR* protocol on these aggregated records in sizes of 7.249, 10.572,

12.674, 17.685, and 23.496. As shown in figure (8), the results indicate the elapsed time to perform (encrypt, calculate ratings and decrypt) with 1024 bits key length. The curve scales linearly as it represents the increase of execution time by increasing the data size.

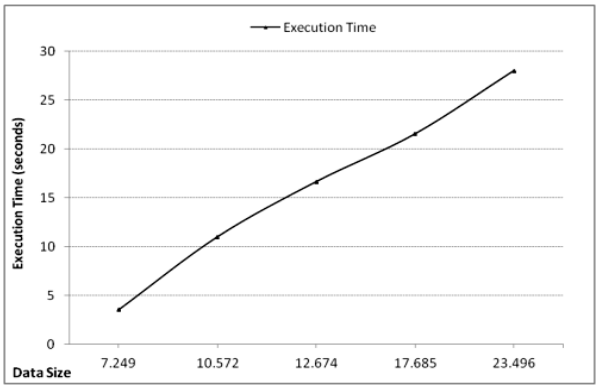


Figure 8: Execution time for different data sizes.

In the first experiment performed on *EVS* algorithm, we measured the relation between different Hilbert curve parameters (order and step length) on the accuracy and privacy levels attained. We mapped the participant's dataset to Hilbert values using orders 3, 6 and 9. We gradually increased the step length from 10 to 80. Figure (9) shows the accuracy of recommendation based on different step length and curve order. We can see that as the order increases, the obfuscated data can offer better predictions for the ratings. Since, with higher values for the curve order, the granularity of the Hilbert curve becomes finer. So, the mapped values can preserve the data distribution of the original dataset. On the other hand, selecting larger step length increases MAE values as large partitions are formed with higher range to generate random values from it, such that these random values substitute real values in the dataset.

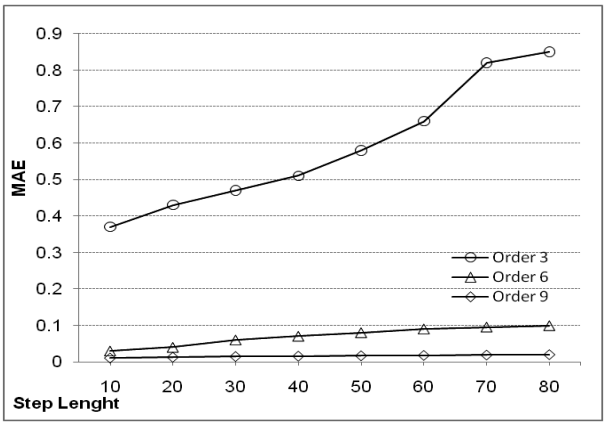


Figure 9: Accuracy level for different step length and orders for *EVS.*

As for the privacy as shown in figure (10), when the order increases a smaller range is calculated within each partition which introduces less substituted values compared with lower orders that attain higher VI values. The reason for this is that larger order divides the m-dimensional profile into more grids, which makes Hilbert curve to better reflect the data distribution. Moreover, we can see that for the same Hilbert curve order the VI values are generally the same for different step length except for order 3, in which VI values has a sharp increase when step length grows from 50 to 60.The effect of increasing step length on VI values is more sensible in lower curve orders as fewer girds are formed and the increase of step length covers more portions of them, which will introduce a higher range to generate random values from it. Based on that, the trust agent employs trust value as an input to tune-up *EVS* parameters in such a way to achieve a trade off between privacy and accuracy.

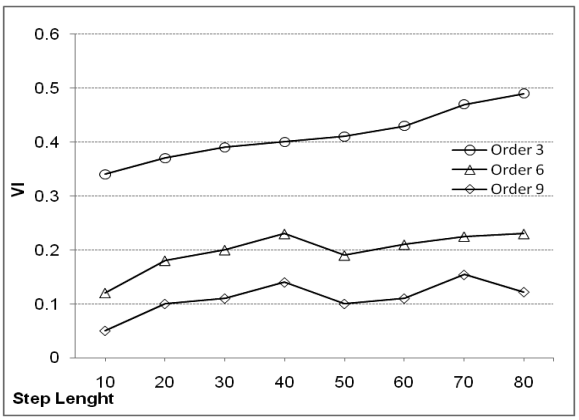


Figure 10: Privacy level for different step length and orders for *EVS.*

We continued our experiments with *EVS* algorithm; we measured the execution time for *EVS* as it is executed locally at the participant's STB box on his profile. The execution time for *EVS* is composed of the time to get partitions based on Hilbert curve and the time to generate random noise. The results for the execution time are shown in figure (11). We can see that as the order of Hilbert curve goes higher, the execution time generally increases than that for a lower order. This growth because of the time consumed in mapping data points to different Hilbert values is dependent on curve order. For different step lengths, the executions time various without substantial trend. As the step length only determines the size of partitions in each dimension; finding these partitions are only dependant on the number of dimensions.

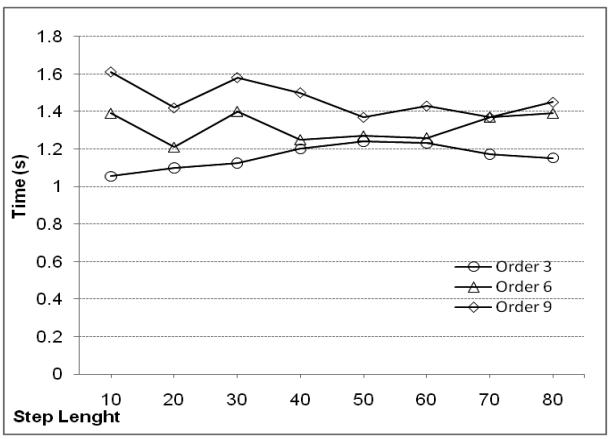


Figure 11: Execution time for different step length in *EVS.*

Finally, we measured the overall recommendation accuracy of our two stage concealment on the same dataset. For *EVS* algorithm, we set the curve order to be 3 (lowest trust level) and the step length to be 10. We first obfuscate different datasets using *EVS* algorithm, then super-peers apply *SR* protocol on these datasets and submit them to PRS. At PRS side, it calculates referrals list then return results back to super-peers which in turn decrypt and publish them. The graph in figure (12) plot MAE values for different data sizes, it clearly shows that the proposed two stage concealment process is very effective in making recommendation and that its privacy preserving nature has marginal impact on the accuracy of recommendation, since *SR* protocol employ homomorphic cryptosystem that preserves the accuracy characteristics of *EVS* algorithm on the dataset. These results indicate two features of the two stage concealment process:

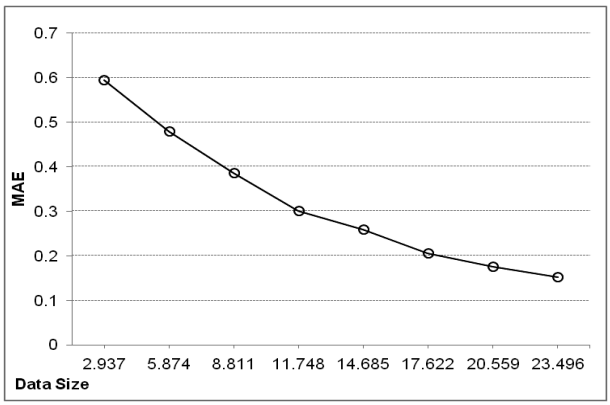


Figure 12: Accuracy of our propsed approch with for different data sizes.

1. Accuracy of the recommendation improves with the increase of collected data, as more diverse ratings produce a reasonable explanation and rank from a reliable sources.
2. Accuracy of the recommendation is reasonable for small datasets, which is highly desirable feature in a dynamic environment like IPTV networks where users' profiles are not large enough.

9 Conclusion and future work

In this paper, we presented our attempt to develop an enhanced middleware for collaborative privacy based on Multi-agent with application to recommender service for IPTV providers. We gave a brief overview of *EMCP* architecture, components and recommendation process. We presented a novel two stage concealment process which provides complete privacy control to participants over their preferences. The concealment process utilizes hierarchical topology, where participants are organized into groups, from which super-peers are elected based on their reputation. Super-peers & PRS use platform for privacy preferences (P3P) policies for specifying their data usage practices. While Participants describe their privacy constraints for the data extracted from their profiles in a dynamically updateable fashion using P3P policies exchange language (APPEL). *EMCP* allows fine grained enforcement of privacy policies by allowing participants to ensure that the extracted preferences for specific request do not violate their privacy by automatically checking whether there is an APPEL preference corresponding to the given P3P policy. Super-peers aggregate the preferences obtained from underlying participants and then encapsulate intermediate values computed on these profiles and then send them to PRS. Trust based obfuscation mechanism is used in the course of participant preferences collection, while the *SR* protocol is used to protect the privacy of collaborative filtering by distributing the participants' preferences between multiple super-peers and encrypting a subset of the aggregated ratings profiles which is useful for the recommendation. We tested the performance of the proposed mechanisms on a real dataset. We evaluated how the overall accuracy of the recommendation depends on data sizes and trust level. The experimental and analysis results show that privacy increases under the proposed middleware without hampering the accuracy of the recommendation. In particular the mean absolute error can be reduced with proper tuning of the trust based obfuscation parameters for a large data sizes. Moreover, utilizing trust levels for obfuscation is an optimization to maintain the utility of the items' ratings. Thus adding the proposed middleware does not severely affect the accuracy of the recommendation based on collaborative filtering techniques.

We realized that there are many challenges in building a privacy enhanced middleware for recommender services. As a result we focused on middleware in a collaborative privacy scenario. A future research agenda will include utilizing game theory to better formulate user groups, sequential preferences release and its impact on privacy of whole profile. We will consider reducing transmission time and the load on the network traffic by adding a secure filtering phase to the *SR* protocol that will allow PRS to exclude items with low predicated rating from the final referrals list. Furthermore it is included to strengthen our middleware against shilling attacks, extending our scheme to be directed towards multi-dimensional trust propagation and distributed collaborative filtering techniques in a P2P environment. Moreover, we need to investigate weighted features vector methods and its impact on released ratings. Such that, the participant not only obfuscates his items' ratings based on the trust level of target-user, but he can also express specific items to be diversely obfuscated with each trust level. We need to perform extensive experiments on other real datasets from the UCI repository and compare our performance with other techniques proposed in the literature. Finally we need to consider different data partitioning techniques as well as identify potential threats and add some protocols to ensure the privacy of the data against those threats.

10 Acknowledgment

This work has received support from the Higher Education Authority in Ireland under the PRTLI Cycle 4 Programme, in the FutureComm Project (Serving

Society: Management of Future Communications Networks and Services).

References

[1] K. Kawazoe, et al., "Platform Application Technology Using the Next Generation Network," NTT2007.

[2] L. F. Cranor, "'I didn't buy it for myself': privacy and Ecommerce personalization," in Designing personalized user experiences in eCommerce, ed: Kluwer Academic Publishers, 2004, pp. 57-73.

[3] M. d. Gemmis, et al., "Preference Learning in Recommender Systems," presented at the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD), Slovenia, 2009.

[4] F. McSherry and I. Mironov, "Differentially private recommender systems: building privacy into the net," presented at the Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Paris, France, 2009.

[5] A. Esma, "Experimental Demonstration of a Hybrid Privacy-Preserving Recommender System," 2008, pp. 161-170.

[6] J. Canny, "Collaborative filtering with privacy via factor analysis," presented at the Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, Tampere, Finland, 2002.

[7] J. Canny, "Collaborative Filtering with Privacy," presented at the Proceedings of the 2002 IEEE Symposium on Security and Privacy, 2002.

[8] H. Polat and W. Du, "Privacy-Preserving Collaborative Filtering Using Randomized Perturbation Techniques," presented at the Proceedings of the Third IEEE International Conference on Data Mining, 2003.

[9] H. Polat and W. Du, "SVD-based collaborative filtering with privacy," presented at the Proceedings of the 2005 ACM symposium on Applied computing, Santa Fe, New Mexico, 2005.

[10] Z. Huang, et al., "Deriving private information from randomized data," presented at the Proceedings of the 2005 ACM SIGMOD international conference on Management of data, Baltimore, Maryland, 2005.

[11] H. Kargupta, et al., "On the Privacy Preserving Properties of Random Data Perturbation Techniques," presented at the Proceedings of the Third IEEE International Conference on Data Mining, 2003.

[12] B. N. Miller, et al., "PocketLens: Toward a personal recommender system," ACM Trans. Inf. Syst., vol. 22, pp. 437-476, 2004.

[13] C.-N. Ziegler, et al., "Improving recommendation lists through topic diversification," presented at the Proceedings of the 14th international conference on World Wide Web, Chiba, Japan, 2005.

[14] J. Golbeck and J. Hendler, "FilmTrust: movie recommendations using trust in web-based social networks," in Consumer Communications and Networking Conference, 2006. CCNC 2006. 3rd IEEE, 2006, pp. 282-286.

[15] A. M. Elmisery and D. Botvich, "An Agent Based Middleware for Privacy Aware Recommender Systems in IPTV Networks," in Intelligent Decision Technologies. vol. 10, J. Watada, et al., Eds., ed: Springer Berlin Heidelberg, 2011, pp. 821-832.

[16] A. Elmisery and D. Botvich, "Private Recommendation Service For IPTV System," in 12th IFIP/IEEE International Symposium on Integrated Network Management, Dublin, Ireland, 2011.

[17] A. Elmisery and D. Botvich, "Agent Based Middleware for Maintaining User Privacy in IPTV Recommender Services," in 3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems, Aalborg, Denmark, 2011.

[18] A. Elmisery and D. Botvich, "Privacy Aware Obfuscation Middleware for Mobile Jukebox Recommender Services," in The 11th IFIP Conference on e-Business, e-Service, e-Society, Kaunas, Lithuania, 2011.

[19] A. Elmisery and D. Botvich, "Privacy Aware Recommender Service for IPTV Networks," in 5th FTRA/IEEE International Conference on Multimedia and Ubiquitous Engineering, Crete, Greece, 2011.

[20] A. Elmisery and D. Botvich, "Enhanced Middleware for Collaborative Privacy in IPTV Recommender Services " Journal of Convergence, vol. 2, p. 10, 2011.

[21] [21] W. Nejdl, et al., "Super-peer-based routing and clustering strategies for RDF-based peer-to-peer networks," presented at the Proceedings of the 12th international conference on World Wide Web, Budapest, Hungary, 2003.

[22] [22] J. Carbo, et al., "Trust management through fuzzy reputation. Int," Journal in Cooperative Information Systems, vol. 12, p. 135—155, 2002.

[23] [23] H. D. Kim, "Applying Consistency-Based Trust Definition to Collaborative Filtering," KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS, vol. 3, pp. 366-374, 2009.

[24] [24] A. Elmisery and D. Botvich, "Agent Based Middleware for Private Data Mashup in IPTV Recommender Services," in 16th IEEE International Workshop on Computer Aided Modeling, Analysis and Design of Communication Links and Networks, Kyoto, Japan, 2011.

[25] [25] D. Kelly and J. Teevan, "Implicit feedback for inferring user preference: a bibliography," SIGIR Forum, vol. 37, pp. 18-28, 2003.

[26] [26] P. Indyk and R. Motwani, "Approximate nearest neighbors: towards removing the curse of dimensionality," presented at the Proceedings of the thirtieth annual ACM

symposium on Theory of computing, Dallas, Texas, United States, 1998.

[27] P. Paillier, "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes."

[28] I. Damgård and M. Jurik, "A Generalisation, a Simpli.cation and Some Applications of Paillier's Probabilistic Public-Key System Public Key Cryptography." vol. 1992, K. Kim, Ed., ed: Springer Berlin / Heidelberg, 2001, pp. 119-136.

[29] I. Damgård and M. Koprowski, "Practical Threshold RSA Signatures without a Trusted Dealer Advances in Cryptology — EUROCRYPT 2001." vol. 2045, B. Pfitzmann, Ed., ed: Springer Berlin / Heidelberg, 2001, pp. 152-165.

[30] D. Boneh and M. Franklin, "Efficient generation of shared RSA keys Advances in Cryptology — CRYPTO '97." vol. 1294, B. Kaliski, Ed., ed: Springer Berlin / Heidelberg, 1997, pp. 425-439.

[31] G. Ghinita, et al., "PRIVE: anonymous location-based queries in distributed mobile systems," presented at the Proceedings of the 16th international conference on World Wide Web, Banff, Alberta, Canada, 2007.

[32] A. Reaz and B. Raouf, "A Scalable Peer-to-peer Protocol Enabling Efficient and Flexible Search," ed, 2010.

[33] R. A. Jarvis and E. A. Patrick, "Clustering Using a Similarity Measure Based on Shared Near Neighbors," IEEE Trans. Comput., vol. 22, pp. 1025-1034, 1973.

[34] K. Liu, et al., "An Attacker's View of Distance Preserving Maps for Privacy Preserving Data Mining Knowledge Discovery in Databases: PKDD 2006." vol. 4213, J. Fürnkranz, et al., Eds., ed: Springer Berlin / Heidelberg, 2006, pp. 297-308.

[35] P.-A. Fouque, et al., "CryptoComputing with Rationals Financial Cryptography." vol. 2357, M. Blaze, Ed., ed: SpringerBerlin / Heidelberg, 2003, pp. 136-146.

[36] D. Gupta, et al., "Jester 2.0 (poster abstract): evaluation of an new linear time collaborative filtering algorithm," presented at the Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, Berkeley, California, United States, 1999.

[37] J. L. Herlocker, et al., "Evaluating collaborative filtering recommender systems," ACM Trans. Inf. Syst., vol. 22, pp. 5-53, 2004.

[38] C. Kingsford, "Information Theory Notes," 2009.

Appendix B: IPTV Recommender Service Scenario

Article VIII

Multi-agent based middleware for protecting privacy in IPTV content recommender Services

Ahmed M. Elmisery, Dmitri Botvich

In the Springer International Journal of Multimedia Tools and Applications (MTAP), Volume 64, Issue 2, May 2013.

Appendix B: Article VIII

Multimed Tools Appl
DOI 10.1007/s11042-012-1067-3

Multi-agent based middleware for protecting privacy in IPTV content recommender services

Ahmed M. Elmisery · Dmitri Botvich

Abstract This work presents our efforts to design an agent based middleware that enables the end-users to use IPTV content recommender services without revealing their sensitive preference data to the service provider or any third party involved in this process. The proposed middleware (called AMPR) preserves users' privacy when using the recommender service and permits private sharing of data among different users in the network. The proposed solution relies on a distributed multi-agent architecture involving local agents running on the end-user set up box to implement a two stage concealment process based on user role in order to conceal the local preference data of end-users when they decide to participate in recommendation process. Moreover, AMPR allows the end-users to use P3P policies exchange language (APPEL) for specifying their privacy preferences for the data extracted from their profiles, while the recommender service uses platform for privacy preferences (P3P) policies for specifying their data usage practices. AMPR executes the first stage locally at the end user side but the second stage is done at remote nodes that can be donated by multiple non-colluding end users that we will call super-peers Elmisery and Botvich (2011a, b, c); or third parties mash-up service Elmisery A, Botvich (2011a, b). Participants submit their locally obfuscated profiles anonymously to their local super-peer who collect and mix these preference data from multiple participants. The super-peer invokes AMPR to perform global perturbation process on the aggregated preference data to ensure a complete concealment of user's profiles. Then, it anonymously submits these aggregated profiles to a third party content recommender service to generate referrals without breaching participants' privacy. In this paper, we also provide an IPTV network scenario and experimentation results. Our results and analysis shows that our two-stage concealment process not only protect the users' privacy, but also can maintain the recommendation accuracy

Keywords Privacy · Clustering · IPTV networks · Recommender System · Multi-agent

1 Introduction

Providers of the next generation of IPTV services seek to gain competitive advantage over competing providers. In order to attract and satisfy customers, they employ automated

A. M. Elmisery (✉) · D. Botvich
Telecommunications Software & Systems Group, Waterford Institute of Technology, Waterford, Ireland
e-mail: ahmedmohmed2001@gmail.com

recommender service to offer added value to their customers [36]. Recommender service collects information about user preferences to create a user profile [27]. The preferences of a user in the past can help the recommender service in selecting items that might be interested for him in the future.

In the current implementations, the recommender services are mostly centralized. The process of generating recommendations in a typical centralized recommender service is based on collaborative filtering technique which is a machine learning technique for filtering information from very large data-sources. Collaborative filtering technique is dexterous enough to recommend new items for users that they might like and they neither consumed nor indicate an interest about them on their profiles before. The collaborative filtering process can be described as follows:

1. The centralized recommender service collects different preference data for all the users. Then it stores these data in its own database.
2. The collaborative filtering technique group users with similar preferences in the same clusters based on specific similarity metric [39]. Then it calculates items with the highest preferences for users within each cluster and recommends them to each user in this cluster.

A closer look to collaborative filtering techniques, we can deduce that building an accurate recommendation model requires end-users to reveal their preference data to the centralized recommender service in order to get useful recommendations. This type of design poses a severe privacy hazard, since the users' profiles are stored in a single entity and under the full control of the host that maintains this recommender service. Moreover, with the advance of clouding computing paradigm, recommender services might operate in countries that have privacy laws different from the country where they are consumed. These variances in privacy laws might not legally bound to ensure users' privacy.

In this work, we present a solution to provide referrals for IPTV users in privacy aware manner, such that it allows the end-user to receive useful recommendations without disclosing their real preferences to that system. The solution is based on AMPR (i.e. acronym for agent based middleware for private recommendations) that maintains user profile privacy in the recommendation process. In the following section we will describe some properties for AMPR:

1. AMPR is running as a multi-agent based middleware to support different types of clients either thin or thick. Moreover, this architecture enables smooth integration with wide range of existing recommender services
2. AMPR [5–7] preserves the aggregates in the obfuscated preferences data to maximize their utility in order to attain acceptable recommendations accuracy, which facilitate AMPR to work with different state-of art filtering algorithms. Extra overhead in computation and communication to be added in the recommendation process due to our two stage concealment process
3. AMPR employs two stage concealment process to conceal the user's preferences in his profile. The real user profile doesn't leave his device until it is properly sanitized and it is maintained encrypted with private password that is known only to the user. If the user doesn't accept to be tracked by the recommender service using his network identity, AMPR hides his identity by routing the submission of his preferences data through relaying nodes in an anonymous communication network before sending it to the recommender service.

In the rest of this paper we will generically refer to news programs, movies and video on demand contents as Items. In section 2 describes some related work. In sections 3 and 4, we introduce our attack model followed by our privacy model. Content recommender service in IPTV network scenario landing AMPR was introduced in section 5. Section 6 introduces our proposed solution. Section 7 goes in the details of privacy breach evaluation. Section 8 describes the recommendation strategy used in PCRS. Section 9 presents some experiments and results based on the proposed solution. Section 10 includes conclusions and future work.

2 Related works

Majority of the existing recommender systems are based on collaborative filtering, building users' profiles in collaborative filtering techniques is done in two ways one is based upon ratings (explicit ratings procedures) and the other is based upon log archives (implicit ratings procedures) [16]. These procedures lead to two different approaches for the collaborative filtering including the rating based approaches and log based approaches. The majority of the literature addresses the problem of privacy on collaborative filtering technique, due to it is a potential source of leakage of private information shared by the users as shown in [30]. In [14] It is proposed a theoretical framework to preserve privacy of customers and the commercial interests of merchants. Their system is a hybrid recommender system that uses secure two party protocols and public key infrastructure to achieve the desired goals. In [1, 2] it is proposed a privacy preserving approach based on peer to peer techniques using users' communities, where the community will have a aggregate user profile representing the group as whole but not individual users. Personal information will be encrypted and the communication will be between individual users but not servers. Thus, the recommendations will be generated at client side. In [34, 35] it is suggested another method for privacy preserving on centralized recommender systems by adding uncertainty to the data by using a randomized perturbation technique while attempting to make sure that necessary statistical aggregates such as the mean don't get disturbed much. Hence, the server has no knowledge about true values of individual rating profiles for each user. They demonstrate that this method does not decrease essentially the obtained accuracy of the results. But recent research work [20, 24] pointed out that these techniques don't provide levels of privacy as it was previously thought. In [24] it is pointed out that arbitrary randomization is not safe because it is easy to breach the privacy protection it offers. They proposed a random matrix based spectral filtering techniques to recover the original data from perturbed data. Their experiments revealed that in many cases random perturbation techniques preserve very little privacy. Similar limitations were detailed in [20]. Storing user's profiles on their own side and running the recommender system in distributed manner without relying on any server is another approach proposed in [31], where authors proposed transmitting only similarity measures over the network and keep users profiles secret on their side to preserve privacy. Although this method eliminates the main source of threat against user's privacy, but it requires higher cooperation among users to generate useful recommendations.

3 Attack model

In our solution, profile is stored at user side in his setup box. Usually user's profile suffers from sparsity problem since it only contains preference data for a small

fraction of items that has been watched by its user. This sparsity in preference data significantly increases the difficulty of any sanitizing techniques as it has an adverse affect on the attained privacy level. Unfortunately, existing algorithms are not effective when applied to sparse preference datasets. Moreover, most of the existing algorithms are built around the assumption that there are two non-overlapping value sets: sensitive values, which need to be kept private, and quasi-identifier values, which can be used by the attacker to identify individuals. While in preference data, these two sets are not disjoint; all preference information could be sensitive, and can also potentially be used as quasi-identifiers. This additional challenge requires new privacy models that need to be applicable for preference data. Our proposed two stage concealment process fall into the category of data obfuscation techniques. As it is known, the attack models for data obfuscation techniques are different from the attack models for encryption-based techniques, but no common standard has been implemented for data obfuscation. Existing techniques has primarily considered a model where the attacker correlates obfuscated data with data from other publicly-accessible databases in order to be able to uniquely identify a user especially if user have "a not-so-popular preferences" then reveal sensitive items in his watching history. In this work, we consider a model where the attacker colludes with some users in the network to obtain some partial information about the process used to conceal the data and/or some of their original preferences data. The attacker can then use this partial information to know a subset of preference data for a certain user or community in the network by matching the partial information he acquired with the released dataset. With call this "reversibility" attack" whereas an attacker can violate the privacy of victims and obtain sensitive information about their sexual orientation or political affiliation.

> Definition 1. Reversibility Attack Given a released dataset O, let's assume O_{ij} is the subset known to the attacker besides concealment procedures. If there are h elements $\in$ O in this subset, we say that preference data for user u can be identified using reversibility attack with probability $\frac{1}{h}$, where h is the size of subset O_{ij}.

So if an attacker knows that Alice involved in recommendation request within a certain community, then this community release its preferences data to PCRS. Knowing the concealment procedures and preferences data revealed from some colluding users. The attacker can match this knowledge to the released data in order to uniquely identify the submitted ratings for items that have been interesting for Alice. Finally, we assuming that both IPTV provider and PCRS follow semi-honest model which is a realistic assumption as both interested in gaining revenues.

4 Privacy model

In the current implementations of recommender services, preferences data shared by participants is assumed to be public information. In most cases this is a reasonable assumption, as user preferences are usually not sensitive in nature. In practical, for some users it is acceptable to release their preferences data to external public parties but others might not willing to do so since they are privacy concerned. As such, we believe that recommender services that require users to divulge their preference data without provable privacy guarantees should be treated with suspicion. In our scenario,

we mitigate this problem by storing the profile at user side such that each STB box maintains a profile that contains preference data for items that have been watched or purchased. When the user releases his preference in a recommendation request, it is required to protect the (lack of) existence of a preference for an item in released profile as this is a sensitive data that must be unavailable throughout the life-time of the recommendation process. Moreover, there should be no link between the participant's identifying information (e.g., name, IP address, etc.) and the released preference data contributed in recommendation process. With these two consolidations, there is still potential threat for participants' privacy as some users in the network might disclose partial information about concealment process or some of their preferences to an attacker in order to divulge the privacy of certain user or their group. In practice, it is hard to predict the amount of partial information that an attacker might gain from this collusion. Therefore, we consider providing protection against the reversibility attack. To achieve this goal, we adapt the definition of MH-Neighbourhood in our model that is analogous to k-anonymity [38]. The conventional k-anonymity model defines quasi-identifier value sets (publicly available information that may be used to identify individuals) and sensitive value sets (private information known only by the individual). These two sets of values are assumed to not overlap. In our problem, all preferences data can be considered both quasi-identifiers and sensitive values. To address this problem, we define MH-Neighbourhood as follows:

> Definition 2. MH-Neighbourhood Given a cluster C_i of original collected preference data, Let $C_i^{'}$ be the obfuscated version of it. We say $C_i^{'}$ satisfy MH-Neighbourhood if there are at least m neighbourhoods, each of which contains h elements such that $C_i^{'}$ is a clustered automorphism of C_i. We say that $C_i^{'}$ is a functional substitute to C_i.

Based on Definition 2, we deduce that $|C_i| = C_i^{'}$ as number and values of elements is equal in both clusters, but elements of C_i is permutated with themselves based on their clustering structure to form $C_i^{'}$. Thus MH-Neighbourhood model is effective for defending against the reversibility attack.

5 Content recommender service in IPTV network- scenario

We extend the scenario proposed in [5–7, 9–12], where a private centralized recommender service (PCRS) is implemented as an external third party server and users give their preferences data to that server in order to receive recommendations. AMPR employs three principles to eliminate the disclosure channels of participants' privacy:

1. At the user side, local obfuscation agent in AMPR applies a certain concealment process on the released preference data before submitting it to super-peers.
2. synchronize agent in AMPR attains anonymity for participant released data in two steps:

   a. Hiding network address (IP) for participant by routing the communication with other parties through relaying nodes in Tor anonymous network
   b. Mixing participant released data with similar data from other participants at the super-peer side.
3. At the super-peer side, global perturbation agent in AMPR applies a certain concealment process on the collected data.

Our two- stage concealment process has two parts, one at participant side and other at super-peer or group leader, so the data doesn't leave the participant side (his STB box) unless it is properly sanitized. The target user receives only the referrals list. In this work, he is not involved in any data collection. Finally, IPTV provider is involved in giving merits to successful super-peers with specific success rates. The super-peers are the ones who communicate with PCRS directly or through the IPTV provider. There is no risk of disclosure from IPTV provider as he will keen to hide hash values for items meta-data, as the offered items considered as business assets to the provider. So both the IPTV provider and PCRS want to cooperate together to offer the required service to end-users but they have conflicting goals. So according to "Nash equilibrium" if PCRS already knows the IPTV provider' choice then no one will gain by changing only his own strategy.

It is difficult to give a clear definition for our two- stage concealment process proposed in the paper, but in general we can define it as: A set of techniques of creating a structurally similar but inauthentic version of users' preferences data that can be used for recommendations purposes. The main aim is to hide the real users' preferences data by making them confusing, willfully ambiguous and harder to interpret for any purpose but in the same time they can act as a functional substitute of real preferences data which is useful only for recommendations purposes. The basic idea for a recommendation based on AMPR is that the user who needs recommendation broadcasts a message to other users in the IPTV network. Then, users who decide to join his recommendation process will form a group with other participants in the IPTV network. The group members elect a highly reputable peer (that we call super-peer or group leader) to mix and collect the preferences they are willing to share from their profiles. Super-peers negotiate with PCRS to express its privacy practices for the data collection and usage via P3P policies which are XML statements that answers questions concerning purpose of collection, the recipients of these profiles, and the retention policy. After receiving P3P policy& request, AMPR ensures that the extracted preferences for specific request do not violate the privacy of its host by checking whether there is an APPEL privacy preference corresponding to that given P3P policy, and then it starts collecting preferences that fulfil the request and in the same time satisfies the extracted APPEL preferences. However, participants cannot trust each other as well and hence the extracted data is sanitized using local obfuscation agent before releasing it to super-peers. Communication between different parties in the group is done through anonymous network in order to hide their network identity. Right after, the super-peer receives preference data, it executes a certain sanitization process using its global perturbation agent in order to conceal the whole group contributions, then super-peer submits the collected data to PCRS in order to produce referrals. The concealed profiles hide the identities of the participants, and thus hamper the ability for the untrusted PCRS to profile and track users.

The Intension to participate is different between users in the same group, as most of users may participate in a request only if they needed referrals about the same category but some of them participant to cooperate in maintaining privacy for the community. The Decision of users to cooperate or not is based on the degree of their contentment with privacy as an essential social norm. Moreover the "super-peer" or the group leader can be motivated to take this role if they know that they might gain some benefits or profits from doing it.

Appendix B: Article VIII

In the following section we will describe some enhancements attained using AMPR:

1. Usage of Pseudonymous for the Profiles: The real user's identity is not always required to provide referrals. Users can be identified by anonymous pseudonyms or nicknames, so that the binding of nickname and the real life identity is not always manifested.
2. User Private Data Store at the Client: Shifting from the approach of storing the user profiles in the server side to the one of storing the profiles on the clients' STBs helps reducing the privacy concerns. One key aspect is keeping the profiles encrypted to avoid people having access to the client's machine or malware that looks for user profiles. The set top box (STB) is an electronic appliance that is equipped with mass storage at the user side. STB connects to both the network and the home television e.g. Cisco STB. A centralized rating database is maintained at PCRS that is used to provide referrals if the number of participants in group fall below a certain threshold.
3. Request-Oriented Collection: Upon receiving a request from the target user, query rewriter and preference checker assures that learning agent extracts only the required preferences from user's profile for a particular request the user is engaged in. The key point relies on knowing what kind of data is required for a given request that can contribute to improve the performance of the recommendation, because the recommender service does not provide recommendation based on one user's full profile information (e.g.: other users' preferences might not be relevant to the request). Likewise, once a user completes a particular request, he/she may no longer be interested in receiving recommendations related to that request for a period.
4. Communication through Anonymous Networks: internet records containing IPs, etc stored at service providers, contain information that permit the identification of user when submitting their preferences to super-peer. AMPR employ anonymous communication to hide the network identity for the participants. The main challenge for AMPR is to tune up and optimize the performance of the anonymous network while maintaining the user anonymity.

Figure (1) shows the architecture of the proposed approach. Additionally the entity operating the recommendation is a third-party recommender service provider employed by the IPTV provider that makes recommendations by consolidating the information received from multiple sources. Our solution relies on the hierarchical topology proposed in [32]; per each request participants are organized into peer-groups managed by super-peers. Electing super-peers is based on negotiation between the participants and security authority centre. The security authority centre (SAC) is a trusted third party responsible for making an assessment on those super-peers according to the participants' reports and periodically updating the reputation of each super-peer based upon it. Reputation mechanisms are employed to elect suitable super-peers based on estimating values for user-satisfaction, trust level, processing capabilities and available bandwidth, further details and information on complex reputation mechanisms can be found in [3]. When a problem occurs with a specific super-peer or participant during the recommendation process, a participant can report it to SAC. After investigation, the assessment of this entity will be degraded. This will limit the chance for electing it as a super-peer in the future. On the other hand, successful recommendation processes will help upgrade the super-peer reputation.

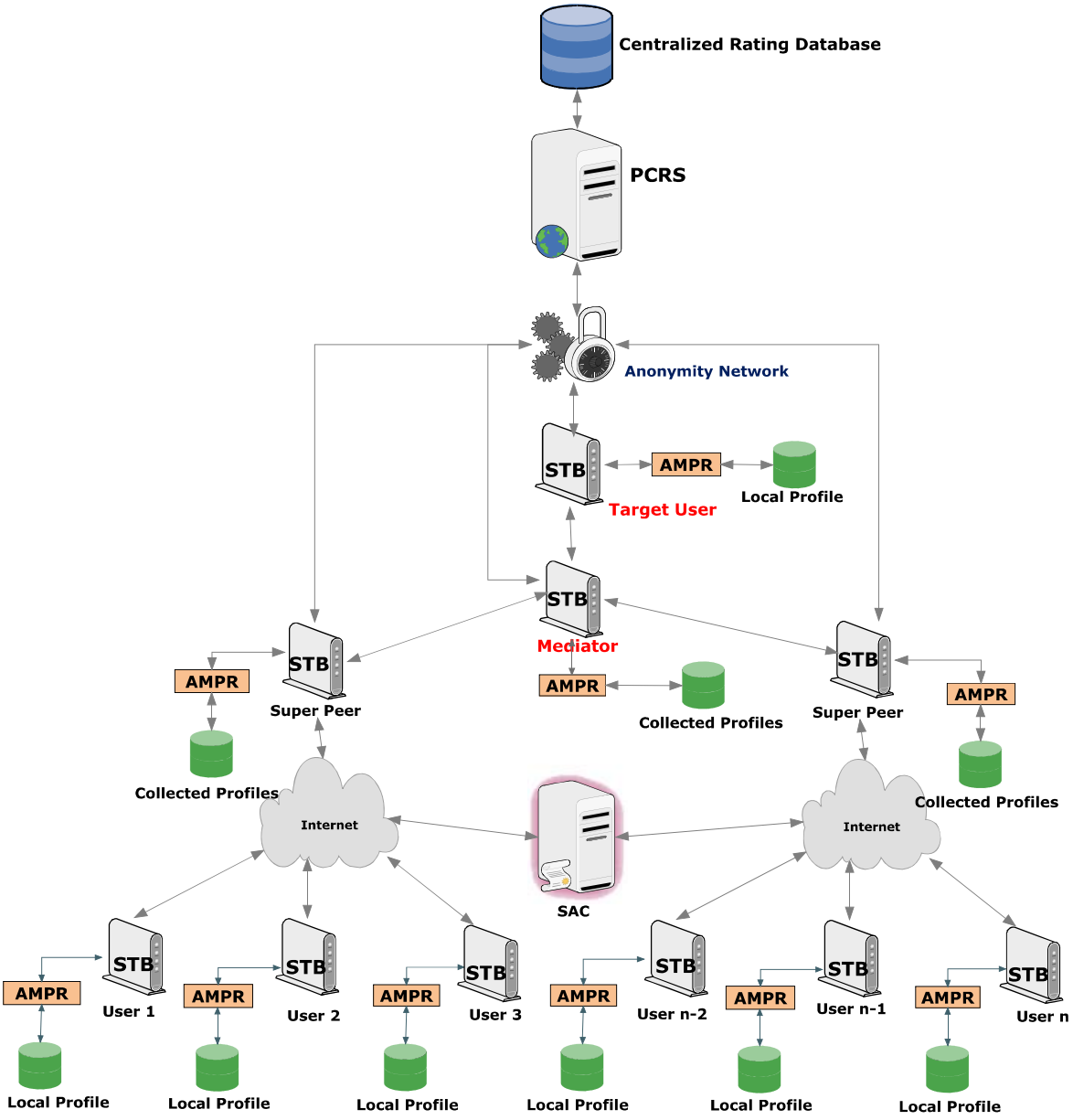


Fig. 1 IPTV Network with Third Party Private Recommender Service

6 Proposed solution

Figure (2) demonstrates AMPR components that are hosted in the STB at the user side and AMPR consists of different co-operative agents. A Learning agent captures user preferences about items explicitly or implicitly to build a rating table and meta-data table. The local obfuscation agent implements *CTA* obfuscation algorithm to achieve user privacy while sharing the data with other users or the system. The global perturbation agent is only invoked if the user is acting as a target user in recommendation process; it executes *ADS*-algorithm on the collected profiles. These algorithms act as wrappers that obfuscate items' preferences before they are fed into the PCRS. Since the database is dynamic in nature, the local obfuscation agent desensitizes the updated data periodically, then synchronize agent send it to the PCRS. So that recommendations to be made on the most recent preferences. More over the synchronize agent will control Tor's routing in order to enhance its performance.

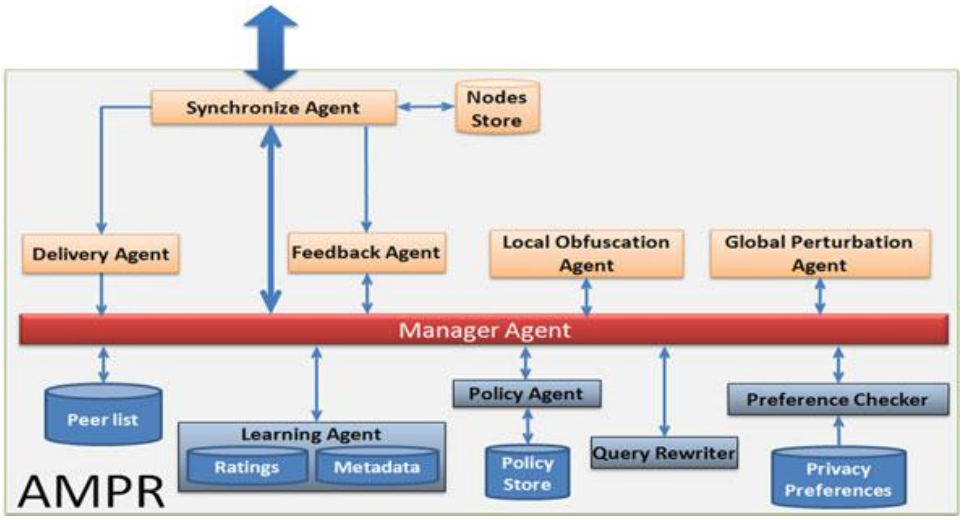


Fig. 2 AMPR Components

In the next sub-sections, we will present our proposed middleware for protecting the privacy of users' preferences. Figure (2) illustrates AMPR components running inside user's STB. AMPR consists of different co-operative agents. A Learning agent that captures user interests about miscellaneous items explicitly or implicitly to build a rating database and meta-data database. The local obfuscation agent implements clustering transformation algorithm (*CTA*) to achieve user privacy while sharing his/her preferences with super-peers or PCRS. The global perturbation agent is only invoked if the user is acting as a super-peer in the recommendation process; it executes advanced data-substitution (*ADS*) algorithm on the collected data. These algorithms act as wrappers that conceal preference data before sharing it with any external entity. Since the database is dynamic in nature, the local obfuscation agent periodically desensitizes the updated preferences, and then a synchronize agent forwards them to the PCRS upon owner permissions. Thus recommendation can be made on the most recent ratings. Moreover, synchronize agent is responsible for calculating & storing parameterized paths in anonymous network that attain high throughput[11], which in turn can be used in submitting preferences anonymously. The policy agent is an entity in AMPR that has the ability to encode privacy preferences and privacy policies as XML statements depending on the host role in the recommendation process. Hence, if the host role as a "super-peer", the policy agent will has the responsibility to encode data collection and data usage practices as P3P policies via XML statements which are answering questions concerning purpose of collection, the recipients of these profiles, and the retention policy. On the other hand, if the host role as a "participant" policy agent acquires the user's privacy preferences and express them using APPEL as a set of preferences rules which are then decomposed into set of elements that are stored in a database called "privacy preferences" as tables called "privacy meta-data". These rules contain both a privacy policy and an action to be taken for such privacy policy, in such way this will enable the preference checker to make self-acting decisions on objects that are encountered during data collection process regarding different P3P policies (e.g.: privacy preferences could include: Certain categories of items should be excluded from data before submission, Expiration of purchase history, Usage of items that have been purchased with the business credit card and not with the private one, Generalize certain terms or names in user's

preferences according to defined taxonomy, Using synonyms for certain terms or names in user's preferences , suppressing certain items from the extracted preferences and insert dummy items that have same feature vector like the suppressed ones as described in [8], limiting the potentially output patterns from extracted preferences etc in order to prevent the disclosure of sensitive preferences in user's profile). Query Rewriter rewrites the received request constrained by privacy preference for its host. The recommendation process in our solution can operates as follows:

1. The learning agent collects the user's interest about different items which represent his profile. The local profile is stored on two databases, the first one is the rating database that contains (item_id, rating) and the second is the meta-data database that contains the feature vector for each item [8] (item_id, feature1, feature2, feature3). The feature vector can include genres, directors, actors and so on. Both implicit and explicit ways for information collection [25] are used to construct these two databases and maintain them.
2. As stated in [5], the target user broadcasts a message to other users in the IPTV network requesting recommendation for a specific genre or category of items. Individual users who have decided to participate in the recommendation process start forming a group. Then, they negotiate with SAC to select a peer with the highest reputation as a "super-peer" which will act as a communication gateway between the PCRS nd the participants in its underlying group.
3. Each super-peer negotiates with both the target user and PCRS to express its privacy policies for the data collection and usage process via P3P policies which are XML statements that answers questions concerning purpose of collection, the recipients of these profiles, and the retention policy.
4. At the participant side, the manager agent receives the request from the target user along with the P3P policy form the elected super-peer; then it forwards P3P policy to preference checker and the request to query rewriter. The preference checker ensures that the extracted preferences for a specific request do not violate the privacy of its host by checking whether there is an APPEL preference corresponding to the given P3P policy and sends it to the query rewriter. The user's preferences can be transferred or collected only if the purpose of statement for the collectors satisfies the privacy preferences. The query rewriter will have knowledge about privacy preferences related to current request via APPEL preference then it rewrites the received request constrained by the privacy preference for its host in order to only retrieve the preferences that the host agrees to share as well as prevent the disclosure of confidential preferences in the participant's profile. This step enable the participant to decide when the recommendation takes place, which information should be collected and for which purpose. Moreover, this step ensures the privacy principles compliance and put the user in control the information that is part of their profiles. The modified request is directed to the learning agent to start collecting preferences that could satisfy the modified query. The manager agent ensures that the collected preferences compliance with the collection data principle, as only the required preferences for the particular request the user is engaged in, is extracted for the local obfuscation process.
5. The local obfuscation agent executes clustering transformation algorithm (*CTA*) on items' ratings that are required in the recommendation process. Moreover the local obfuscation agent hashes their identifiers and meta-data using locality-sensitive hashing

(LSH) [22] to hash these values. One interesting property for LSH is that similar items will be hashed to the same value with high probability. Super-peers and PCRS are still able to perform computation on the hashed values using appropriate distance metrics like hamming distance or dice coefficient.

6. Participants submit their obfuscated preferences to the super-peers of their group. Anonymous communication [11] utilized to hide the network identities of group members when submitting their obfuscated preferences to the super-peers. Finally, the policy agent audits the original and modified requests plus P3P policy with previous requests; this step allows AMPR to prevent multiple requests that might extract sensitive preferences. In such a case, if the target user requests same data twice, level of obfuscation in the extracted preferences is increased such that extracted preferences appear as a completely different set of preferences to the target user.
7. Upon receiving the obfuscated preferences from the participants, each super-peer collects the participants' pseudonyms and builds a group profile such that all the < hashed value, rating > elements belonging to similar items are grouped together. Then super-peer incites his global perturbation agent to perturb the collected profiles. Finally. The super-peer can seamlessly interact with the PCRS by posing as an end-user and has a group profile as his own profile in order to attain recommendations.
8. PCRS performs its filtering techniques on the group profile which in turn return a list of items that are correlated with these profiles. The list can be encrypted with a private key provided by super-peer. The list is send back on the reverse path to the super-peer that in turn decrypt and publish it anonymously to the other group users that participated in the recommendation process. Finally, each participant report scores about the elected super-peer of his group to SAC, which helps to determine reputation of each entity involved in referrals generation.

7 Anonymity for participants' submissions

In our solution, the participants had to submit their locally obfuscated data to super-peers of their group. In order to ensure that there is no association between a participant's identifying information and his/her released data, all submissions are made anonymously. That is, participants use a system for anonymous routing to submit their contribution to a super-peer that will receives all preference data, then submits it to PCRS after performing the second concealment process (global perturbation). Anonymous networks are readily available today such as, Tor anonymity network [4]. With this design, we can achieve the required privacy of a participant who submits his /her preference data is not compromised because the super-peer learns no information about the user's identity. Alternatively, participants can use this anonymous network in retrieving the referrals list. In this work, we employed Tor network for that purpose; TOR includes a small set of trusted authoritative directory servers responsible for aggregating and distributing signed information about known routers in the network. Tor clients periodically fetch the directory information from directory mirrors in order to learn information about other servers in the network, such as their IP addresses, public keys, etc. Tor network suffers from serious performance degradation because of its random path selection algorithms. The number of

Tor routers is the path length and the client negotiates session keys with the chosen routers to form a circuit. Other path selection algorithms in Tor use self-reported bandwidth values that might select with high probability a router with low bandwidth because it is sensitive to loads and changing network conditions. The synchronize agent enhances the current path selection algorithms in Tor to improve its performance during communicating other nodes in the recommendation process. The synchronize agent partition the Tor network into classes of high or low bandwidth Tor routers. Paths drawn from the class of high-bandwidth routers can provide better performance. Paths can be reserved for participating in specific recommendation requests based on a user's priorities or preferences. Therefore, inspired from the work in [21, 33] we have implemented a simple parameterized path selection algorithm (PPS) that allow the synchronize agent to enhance the path selection in the Tor network with two priorities and it can be easily extended to support priorities larger than two. The synchronize agent can create circuits in advance to reduce the waiting time then measure P_T before using each path. PPS consists of the following steps:

1. The user input minimum path throughput P_T, and circuit throughput C_T to the synchronize agent.
2. Based on Tor authoritative directory servers, the algorithm start partitioning the class of high-bandwidth routers into a set of overlapping clusters (based on geographical location, platform, bandwidth, uptime, last update, number of connections and self-reported bandwidth estimate) using the algorithm on [15].
3. It builds a pool of Tor nodes whose bandwidth $\geq P_T$ from each cluster. In order to decrease the delay in circuit creation, the synchronize agent can select overlapping routers between clusters
4. Then, it randomly builds circuits passing through these clusters. Then measure each circuit throughput, and select the first circuit that achieve bandwidth $\geq C_T$.
5. The synchronize agent then negotiates session keys with each router in the circuit. It first starts with the entry router in the sequence. Then the connection to the middle routers is done via the encrypted tunnel established with the antecedent routers, and then again with the exit router in the circuit. The exit router is responsible for establishing the connection from the Tor network to the client's intended destination
6. The synchronize agent records the previously used Tor nodes and exclude them from future circuit building clusters.

One important aspect that should be kept in mind, the anonymous nature of participations can make the system prone to abuse. That is, dishonest users might attempt to influence the recommendation process by submitting false or repeated preferences. To combat this issue, we employ SAC to manage users, such that all users are required to register themselves in SAC prior using recommender service. Then, SAC will start issuing them anonymous credentials at the time of registration. PCRS can authenticate the submission using anonymous credentials issued by the SAC. By using a type of anonymous authentication it allows to indentify the malicious participants under exceptional circumstances, so the abuse of PCRS can be mitigated.

8 Proposed obfuscation algorithms

In the next subsections, we present two different algorithms used by the obfuscation agents in AMPR to fulfil our two stage concealment process. The extracted data is

concealed in a way to preserve user's privacy in recommendation process with minimum loss of accuracy. We perform experiments on real datasets to illustrate the applicability of the proposed algorithms and the privacy and accuracy levels achieved using them.

8.1 Local obfuscation using clustering transformation algorithm (CTA)

We propose a novel algorithm for obfuscating the user profile before sharing it with super-peer in the IPTV network. This algorithm called *CTA*, which has been designed especially for the sparse data problem we have here. We noted that, the available anonymisation algorithms increase data distortion and, as result inaccurate recommendation model could constructed. Maintaining utility and privacy for profiles seems to be contradictory goals to attain. The key idea for *CTA* is based on the work in [17] that uses Hilbert curve as a dimensionality reduction tool to create a cloaking regions to attain privacy for users. Hilbert curve also has the ability to maintain the association between different dimensions. In this subsection, we extend this idea as following, we also use Hilbert curve to map m-dimensional profile to 1-dimensional profile then *CTA* discovers the distribution of that1-dimensional profile. Finally, we perform perturbation based on that distribution in such a way to preserve the profile range to provide high accurate results when performing recommendations. The output of the proposed obfuscation algorithm should satisfy two requirements:

1. Reconstructing the original profile from the obfuscated profile should be difficult, in order to preserve privacy.
2. Preserve the distances of the data to achieve accurate results for the recommendations.

The steps for *CTA* algorithm consists of the following steps:

1. We denote the collected m-dimensional user profiles as dataset D of c rows, where each row is a sequence of m dimensions $A = A_1, A_2, A_3, A_4\ldots\ldots, A_m$.
2. *CTA* divides the m-dimensional profile into grids of order k (where k is user defined value) as shown in [17, 37]. For order k, the range for each dimension divided into 2^k intervals.
3. For each dimension $\forall_{i=1}^{m} A_i$ of the collected profile D:

   a. Compute the k-order Hilbert value for each data point $\forall_{x=1}^{c} a_{ix}$. This value represents the index of the corresponding interval where it falls in.
   b. *CTA* sort the Hilbert values from smallest to biggest, then use the step length (a user defined parameter) to measure whether any two values are near from each other or not. If these values are near, they are placed in the same partition $\forall_{v=1}^{k} k_{iv}$.

   These two steps iterates for all m-dimensions. The final result from these steps is k partitions for each dimension denoted as $\forall_{i=1}^{\mathbf{m}} \forall_{v=1}^{\mathbf{k}} \mathbf{C_{iv}}$
4. *CTA* constructs a N shared nearest neighbour sets S_r *where* r=1....N as in [23] from different partitions with a new modified similarity function as following, two partitions in different dimensions C_{iv}, C_{i+1v} form a shared nearest neighbour set S_r if they share k-number of common elements such that $S_r = C_{iv} \cup C_{i+1v}$
5. For each newly created set S_r, *CTA* calculates its interquartile range. Then, for each point $a_i \in S_r$ generate a uniform distributed random point n in that range that can substitutes a_i.
6. Finally, the new set $D' = \cup_{r=1}^{N} S_r$ is sent to PCRS.

8.2 Global perturbation using advanced data-substitution (ADS) algorithm

After executing the local obfuscation process the global perturbation using *ADS* phase starts. *ADS* is a lossless algorithm that preserves the privacy of the collected preferences data without loss of information content for recommendation process, maintains correlation between various dimensions in the data and allows extracting inferences on the data in private way; these properties make it more suitable for outsourcing scenarios. *ADS* algorithm can be performed on any data set in metric space that has a distance measure between its elements. The complexity of *ADS* is function of number of data elements in the neighbourhoods and size of participants' community. *ADS* algorithm does not require the whole dataset to be retained in the memory since it can be applied to limited number of clusters per time, which is demonstrated later using the execution time required for *ADS* on large dataset. *ADS* algorithm partitions the collected preferences data into groups then permutes data elements in each group within each other based on the gradually ascent nature of each group; furthermore it permutes the whole group with another neighbouring group similar to it. *ADS* algorithm begins with clustering the preference data elements together using LLA algorithm [13] to form neighbourhoods, then each neighbourhood is ordered in density structure based on distance from the local core point in ascending manner. Moreover, algorithm precedes to merge entire neighbourhood with another as long as the error of merging them err ≤UD is under specific threshold specified by the user otherwise it substitutes the entire neighbourhood with another as if they are reachable to each other through number of steps defined by parameter ω, that we call step length. The collected preferences data are divided into a set of clusters C_i, $i \in [1, k]$ using LLA algorithm that represent the input to *ADS* algorithm. Each cluster C_i consists of n records with m attributes, and C_i', $i \in [1, k]$ represents the perturbed database. Each cluster consists of a group of dense regions (neighbourhoods) with local core point; the extracted neighbourhoods are likely to have same size that is a parameter defined by the user for LLA algorithm, so that each neighbourhood has at least a number of data elements to hide it. *ADS* algorithm is a recursive algorithm that is performed individually on each cluster using a tree traversal approach to arrange and permutes dense regions inside clusters. The create_tree procedure builds a tree for each cluster C_i, such that the global core-point is the root of tree and the nodes of the tree correspond to local core-points for each neighbourhood, all leaf nodes (data elements) point to their local core-point in the neighbourhood. The data elements within neighbourhoods are ordered from left to right based on their distance from their local core-points. The distance between each data element and local core-point are represented by the edge connecting them. The procedure ancest (TLUT, node[j]) returns all the ancestors of given node, and the procedure merage(c_{ia}', c_{ib}') merges two perturbed neighbourhood such that each node in the second tree is going to be merged into the first tree. The merging process start by creating edge between local core-point in second tree to the root node in the first tree, and update all nodes in second tree to make them all point to the root node of the first tree. child (TLUT) holds the set of valid children for a specific local core-point. The created tree is then stored in variable $tree_j$, then the size for each tree is calculated and stored in d_j, then d_j is sorted such that the final perturbed database C_i' is formed based in this sorted list. This substitution structure obtained using *ADS* algorithm is chosen as a replacement for the original data set. The transformation performed using *ADS* on preference data protect it from the reversibility attack. *ADS* algorithm consists of the following steps:

Appendix B: Article VIII

Input: $C_i, \ldots \ldots C_k$

Output: $C_i^{'}, \ldots \ldots C_k^{'}$

For each C_i **where** $i = [1 \ldots k]$

C_i =Sort $\left(c_{ij}, \ldots \ldots c_{im}\right)$

For each c_{ij} **where** $j = [1 \ldots m]$

$tree_j = \textbf{create\_tree}\left(c_{ij}, 0\right)$

$d_j = \textbf{Enum}\left(tree_j\right)$

For each $c_{ia}^{'}, c_{ib}^{'}$ **where** $a, b \in j = [1 \ldots m]$ and $a \neq b$

If $err = \frac{c_{ia}^{'}.w_{ia} \times c_{ib}^{'}.w_{ib} \times d\left(c_{ia}^{'}, c_{ib}^{'}\right)^2}{c_{ia}^{'}.w_{ia} + c_{ib}^{'}.w_{ib}} \leq UD$ **then Merage**$\left(c_{ia}^{'}, c_{ib}^{'}\right)$

ElseIf $c_{ia}^{'}$ is within ω steps reachable to $c_{ib}^{'}$ **then** substitute $c_{ia}^{'}$with $c_{ib}^{'}$

d_j= **Sort**$\left(d_j\right)$

For each d_j **in the** $tree_j$

Select $c_{ij}^{'}$ **with** d_j

$final\_set = final\_set \cup c_{ij}^{'}$

$C_i^{'} = finalset$

Creat_tree$(node[j], TLUT)$

If $TLUT = 0$ **then** $TLUT =$ **globalcorepoint**(C_i)

If $node[j] = \emptyset$ **then return** $TLUT$

Select $node[j]$**such that** ***dist*** $(node[j], node[j-1]) = minimal$

$TE = node[j] -$ **ancest**(TLUT, $node[j]$)

child$(TLUT)$= **sort**(TE)

$TLUT =$ **child**$(TLUT)$

$$w_{ij}\left(localcorepoint(c_{ij})\middle|x_1, x_2, .. x_o\right) = \sum_{e=1}^{o} \frac{\left\| f^{x_n}\left(localcorepoint(c_{ij})\right)\right\|^2}{\left\|x_e - localcorepoint(c_{ij})\right\|^2}$$

merage$\left(c_{ia}^{'}, c_{ib}^{'}\right)$

If localcorepoint$\left(c_{ia}^{'}\right) >$ **localcorepoint**$\left(c_{ib}^{'}\right)$ **then**

$TLUT =$ **localcorepoint**$\left(c_{ia}^{'}\right)$, M_child$\left(c_{ib}^{'}\right)$

Else $TLUT =$ **localcorepoint**$\left(c_{ib}^{'}\right)$, M_child$\left(c_{ia}^{'}\right)$

createdge (A) **such** A = ***dist*** $\left(\textbf{\textit{localcorepoint}}\left(c_{ia}^{'}\right), \textbf{\textit{localcorepoint}}\left(c_{ib}^{'}\right)\right)$

For each $element[j]$ **in** M_child

createdge (j) **such** $j =$ ***dist*** $(TLUT, element[j])$

Update $element[j]$ **such that** parent= M_ancest

$d_{Mer} = d_a + d_b$

9 Privacy breach evaluation

The proof of privacy for both CTA and ADS algorithms depends on how much information is leaked during the execution of the recommedation process. At the same time, our proposed algorithms should output accurate results.

9.1 Privacy breach evaluation for CTA algorithm

Privacy breach can be described in terms of how well the original user's ratings can be estimated from the submitted obfuscated ratings. Unlike other techniques, our method generates new data points, whose interpoint distances approximate the original distances. Consequently, points which lie close to one another in the original space mostly remain close to each other in the transformed space. Therefore, it seems theoretically to be more resilient to some potential attacks [29] that exploit the properties of the released data. These attacks are based on how much information about original data is available to the attacker that is obtained through either known input-output and known sample. In the known input-output, attacker knows collection of linearly dependant original data points and points they map in perturbed data. While in known sample, assumes that original data arose as independent samples of multidimensional random vector with unknown probability density function, and the attacker has access to a collection of these independent samples. In CTA algorithm, the linear ordering based on Hilbert curve retains the proximity and neighboring aspects of the original data. We define H_d^N for $N \geq 1$ and $d \geq 2$ as the Nth order Hilbert curve (user defined values) for a d- dimensional space. $H_d^N : [0, 2^{Nd} - 1] \rightarrow [0, 2^N - 1]^d$ as follows Hilbert value $H = \in (P)$ for $H \in [0, 2^{Nd} - 1]$, where P is coordinate of each point in $[0, 2^N - 1]^d$. Thereafter, we cluster nearby Hilbert values based on step length (a user-defined parameter) then CTA substitutes each point in the group with uniform distributed random point in the same interquartile range for that cluster. Therefore we can consider as a one-way function if the curve parameters are unknown. These parameters include (starting point, N, step length) are defined at the participant side and any external entity only know the final perturbed data that participant agree to release. As a result, the statistical information from the perturbed data are inconsistent with that from the original data. Therefore, attacks such as those described before would be in efficient in breaching privacy. In addition to that, clustering Hilbert values and substituting each point with random point introduces uncertainty about exact distance between data points, thus will make any distance based attach ineffective.

9.2 Privacy breach evaluation for ADS algorithm

In this section, we analyze the guarantee that *ADS* algorithm provides protection against reversibility attack. Reversibility In normal data substitution techniques is mainly depend on minimum number of records sufficient to obtain original dataset; that is here regard as the half number of records in dataset since for all substituted items, other items have to be revealed. In the case of *ADS* algorithm, a complete or partial reversal of the dataset requires the knowledge of neighbourhood size m, number of global core-points k, merging error err and step length ω. As the goal of the attacker is to obtain the original value that is equals to obfuscated value with a higher degree of certainty.

Theorem 1 Assume [*O,B*] be the original and substituted preferences data of size *n* respectively

$$O = o_1, \ldots\ldots\ldots o_n$$
$$B = b_1, \ldots\ldots\ldots b_n$$

If we assume that attacker want to reveal substituted elements b_{ijx}, b_{ijy},$b_{ijz} \in B$ which are corresponding to original elements o_{ijx}, o_{ijy}, $o_{ijz} \in O$ that belong to neighbourhood c_{ij} of size h. Moreover, let's assume that all elements in c_{ij} are distinct. So if the attacker knows *ADS* algorithm, including neighbourhood size h and a subset of O with no prior information employing reversibility attack. Then, an attacker cannot know the participant ratings with confidence at most $\frac{1}{\mathrm{h}}$ as he needs to know at least $h - 3$ elements exist within c_{ij}.

Proof let [O_{ij}, B_{ij}] be the original and substituted data items in neighbourhood c_{ij} of size h. Such that $O_{ij} = o_{ij1}$,.....o_{ijh} and $B_{ij} = b_{ij1}$,.....b_{ijh}, we want to determine the effect of reversibility attack with knowledge of $h - 3$ original elements. We assume the only information revealed to attacker are set of $h - 3$ original data items $O'_{ij} = o_{ij1}, \ldots\ldots\ldots o_{ijh-3}$ and $B = b_1$,.....b_n. The goal of attacker is to reveal original values of b_{ijx}, b_{ijy}, $b_{ijz} \in B_{ij}$ that are equal to o_{ijx}, o_{ijy} and o_{ijz}. In such way, we have two cases first case: elements b_{ijx}, b_{ijy},$b_{ijz} \in B$ falls within the set $\{b_{ij1}, \ldots\ldots\ldots b_{ijh}\} \in c_{ij}$. However, he can't know their exact locations, since there are many placements for b_{ijx}, b_{ijy}, b_{ijz} inside c_{ij}. If we assume a worst case for breach, so attacker knows $r \leq h$ elements (a reasonable assumption due to the sparsity of recommender datasets), he still can't reveal the original values for these elements. The reason is that order of elements in neighbourhood may be changed due to merging or substituting operations performed between whole neighbourhoods. In case of all of that are available to the attacker, only values inside neighbourhood and other merged or substituted one (if any) will revealed, without breaching values for other neighbourhoods on same cluster or inside other clusters.

Since there are no additional information revealed about number of global core-points k, merging error err and step length ω. The attacker can't determine accurately their original values inO'_{ij}. Second case: elements b_{ijx}, b_{ijy},$b_{ijz} \in B$ do not falls in neighbourhood c_{ij}. In this case, these elements can belong to any neighbourhood c_{xy} of size h where y = [1…m] exists within any cluster C_x where x = [1…k–1]. The attacker will not be able to determine at all their original values inO'_{ij}. This proves that knowing h elements in the neighbourhood will not assist attacker to determine original values for the remaining neighbourhoods.

Theorem 2 Privacy for participants' ratings is attained using *ADS* algorithm. If we assume that all ratings are independent. Then an attacker with prior information $b \leq \frac{1}{\mathrm{h}}$ for item x employing a reversibility attack cannot predict participant's rating with confidence greater than $\frac{1}{h} + \frac{b}{2-hb}$

Proof Assume that an attacker using reversibility attack managed to identify neighbourhood c_{ij} containing ratings for user u. Based on the existence of a rating for item x in the released dataset, the attacker would like to know if item x is rated by user u. With no additional information revealed about number of global core-points k, merging error err and step length ω. The attacker can only infer that user u had rated x with probability at least $\frac{1}{\mathrm{h}}$. Let $p(u,x)$ be the probability that user u rated item x in his released preferences. And let $p(c_{ij}, x)$ be the unconditional probability that at least one user in neighbourhood c_{ij} rated item x. We wish to calculate $p\big((u,x)|(c_{ij},x)\big)$ which is the probability that user u rated item x such that the rate to that item still exists in the neighbourhood c_{ij}. Using bayes' rule, we can have

$$p\big((u,x)|(c_{ij},x)\big) = \frac{p\big((c_{ij},x)|(u,x)\big) * P(u,x)}{p(c_{ij},x)} \tag{1}$$

From *ADS* algorithm we can say that$p\big((c_{ij},x)|(u,x)\big) = 1$, $p(u,x) = b$ since we assume that each user u rated item x independently with probability b. Then we can find a bound on $p(c_{ij}, x)$ as follows:

$$\begin{aligned} p(c_{ij},x) &= 1-(1-b)^h \\ &= 1-\left(1-hb+\binom{h}{2}b^2-o(b^3)\right) \\ &= hb-\binom{h}{2}b^2-o(b^3) \geq hb-\binom{h}{2}b^2 \end{aligned}$$

Combining this with Eq. 1, we get

$$\begin{aligned} p\big((u,x)|(c_{ij},x)\big) &\leq \frac{1.b}{hb-\binom{h}{2}b^2} \\ &= \frac{1}{h}\left(\frac{hb}{hb-\binom{h}{2}b^2}+\frac{-\binom{h}{2}b^2}{hb-\binom{h}{2}b^2}\right)+\frac{1}{h}\left(\frac{\binom{h}{2}b^2}{hb-\binom{h}{2}b^2}\right) \\ &= \frac{1}{h}+\frac{1}{h}\left(\frac{\frac{1}{2}h(h-1)b^2}{hb-\frac{1}{2}h(h-1)b^2}\right) \\ &= \frac{1}{h}+\frac{1}{h}\left(\frac{(h-1)b}{2-(h-1)b}\right) \leq \frac{1}{h}+\frac{b}{2-hb} \end{aligned}$$

10 Recommendation strategy

In This work, we will test the proposed algorithms in one of the online mode recommendation algorithm, further discussion for offline mode will be presented in future work. Online mode algorithms run at PCRS and attempt to make predictions on the ratings of a particular user by collecting preference information from other users. The collected profiles (group profile) represented as m*n user item matrix which contains a collection of numerical ratings of M users on N items. After that, the neighbourhood formation at PCRS is done by calculating the similarity between users in the user-item matrix. Users similar to the referrals requester using some proximity metric will form a proximity based neighbourhood with him [34]. This neighbourhood will be utilized later for predication step. The prediction on rating of user i for item k is given by a weighted average [28] of users whose ratings are similar to the target user.

$$P_{ik} = \overline{v}_i + \frac{\sum_{j\in U_k} s(u_i,u_j)\left(v_{jk}-\overline{v}_j\right)}{\sum_{j\in U_k}\left|s(u_i,u_j)\right|}$$

Where$U_k = \{i \in U | v_{ik} \neq \emptyset\}$ is the set of users who have rated k–th item. $\overline{v_j}$ is the mean of all ratings made by user i. The weights of average $s(u_i,u_j)$ are the similarity between user u_i and u_j such as the Pearson correlation coefficient or Euclidean distance. We represent the user as a vector consists of n features slots, one for each item. These slots contain user's

ratings for different items or ∅. The similarity between users' vectors is calculated as the cosine of the angle formed between them as following:

$$s(u_i, u_j) = \frac{\sum_{k=1}^{n} v_{ik} v_{jk}}{\sqrt{v_{i1}^2 + \ldots + v_{in}^2}\sqrt{v_{j1}^2 + \ldots + v_{jn}^2}}$$

Finally, the recommendation process is to produce a predicted rating based on that neighbourhood for a list of items that have not been rated by the user, these items have a high potential of interest (predicated with high positive rating) to the user. The resulting items list can be further constrained based on marketing or Qos rules.

11 Experiments

The experiments are run on 2 Intel® machines connected on local network, the lead peer is Intel® Core i7 2.2 GHz with 8 GB Ram and the other is Intel® Core 2 Duo™ 2.4 GHz with 2 GB Ram. We used MySQL as data storage for the users' preferences that acquired by learning agent. The proposed two stage concealment process is implemented in C++ using the MPICH implementation of the MPI communication standard for distributed memory implementation of the *ADS* algorithm to mimic a distributed reliable network of peers. The experiments presented here were conducted using the Jester dataset provided by Goldberg from UC Berkley [18]. The dataset contains 4.1 million ratings on jokes using a real value between (−10 and +10) of 100 jokes from 73.412 users. The data in our experiments consists of ratings for 36 or more items by 23.500 users. We evaluated the proposed solution from different aspects: privacy achieved accuracy of results and performance. We used the mean absolute error (MAE) metric proposed in [19]. MAE is one of most famous metrics for recommendation quality. As it measures the predication verity between the predicated ratings and the real ratings, so smaller MAE means better recommendation provided by PCRS. We can define it as following: Given a user predicated ratings set $p = \{p_1, p_2, p_3 \ldots p_N\}$ and the corresponding real ratings set $r = \{r_1, r_2, r_3 \ldots .r_N\}$, MAE is:

$$MAE = \frac{\sum_{i=1}^{N} |p_i - r_i|}{N}$$

To measure the privacy or distortion level achieved using our proposed algorithms, we used the variation of information metric VI [26] to estimate data error. A higher value of VI means a larger distortion between the obfuscated and original dataset, which means a higher level of privacy.

$$VI = H(p) + H(r) - 2I(p, r)$$

Here H(p) is entropy of p, r and I(p,r) is mutual information between p and r. The experiments involve dividing the data set into a training set and testing set. The training set is concealed then used as a database for PCRS. Each rating record in the testing set is divided into rated items t_i and unrated items r_i. The set t is presented to PCRS for making predication p_i for the unrated items r_i. The experiments were performed while keep the number of super-peers n=9, as described earlier they will be responsible for aggregating the data of 23.496 participants. The recommendation process can be initiated by any user that will act as a requester for the referrals list.

In the first experiment performed on *CTA* algorithm, we measured the relation between different Hilbert curve parameters (order and step length) on the accuracy levels attained. We map the locally obfuscated dataset to Hilbert values using order 3, 6 and 9. We gradually increased the step length from 10 to 80. Figure (3) shows the accuracy of recommendations based on different step length and curve order. We can see that as the order increases, the obfuscated data can offer better predictions for the ratings. This is because as the order has higher value, the granularity of the Hilbert curve becomes finer. So, the mapped values can preserve the data distribution of the original dataset. On the other hand, selecting larger step length increases MAE values as large partitions are formed with higher range to generate random values from it, such that these random values substitute real values in the dataset. As for the privacy as shown in Fig. (4), when the order increases a smaller range is calculated within each partition which introduces less substituted values compared with lower orders that attain higher VI values. The reason for this is larger order divides the m-dimensional profile into more grids, which makes Hilbert curve better reflects the data distribution. Also, we can see that for the same Hilbert curve order the VI values are generally the same for different step length except for order 3, in which VI values has a sharp increase when step length grows from 50 to 60. The effect of increasing step length on VI values is more sensible in lower curve orders as fewer girds are formed and the increase of step length covers more portions of them, which will introduce a higher range to generate random values from it. So the participant should select *CTA* parameters in such a way to achieve a trade off between privacy and accuracy.

We continued our experiments with *CTA* algorithm; we measured the execution time for *CTA* as it is executed locally at the participant's STB box on his profile. The execution time for *CTA* is composed of the time to get partitions based on Hilbert curve and the time to generate random noise. The results for the execution time are shown in Fig. (5). We can see that as the order of Hilbert curve goes higher, the execution time generally increases than that for a lower order. This growth because of the time consumed in mapping data points to different Hilbert values is dependent on curve order. For different step lengths, the executions time various without substantial trend. As the step length only determines the size of partitions in each dimension; finding these partitions is only dependant on the number of dimensions.

For *ADS* algorithm, in order to measure the distortion in prediction accuracy regarding different neighbourhood sizes, we carried out predication phase for various neighbourhood

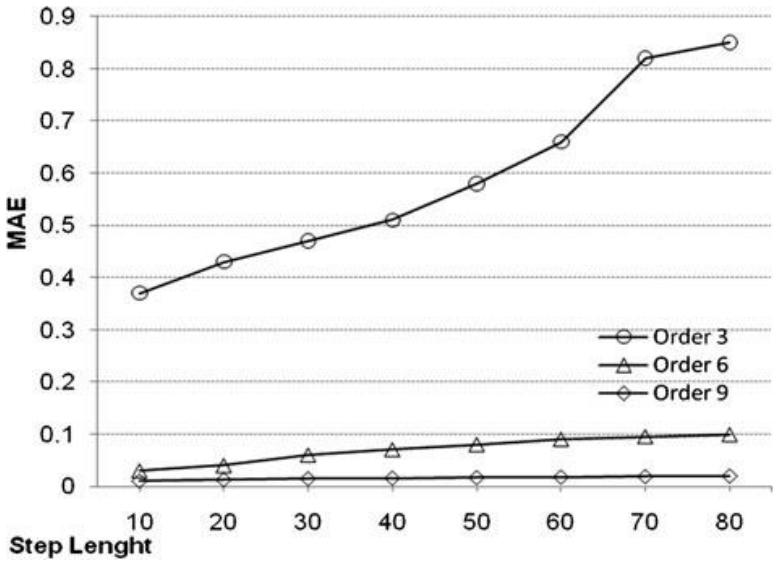


Fig. 3 Accuracy level for different step length and orders for *CTA*

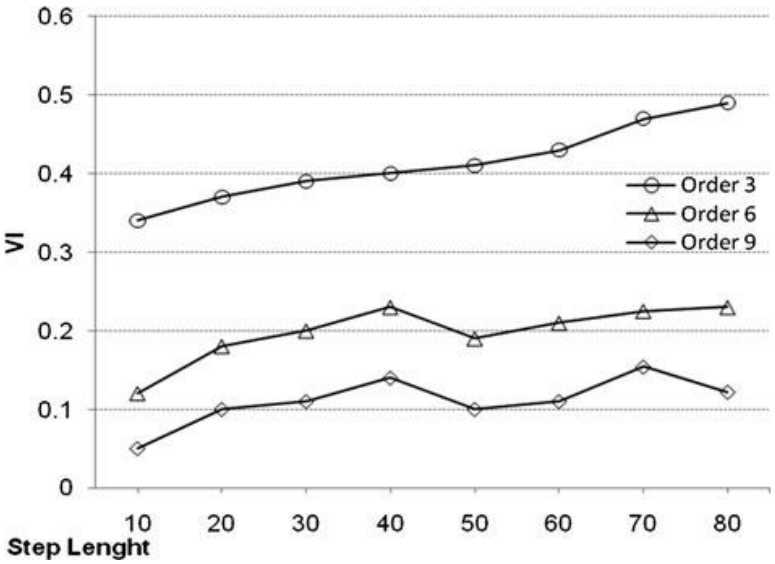


Fig. 4 Privacy level for different step length and orders for *CTA*

sizes varying from 1 % of cluster size to 50 % of cluster size. The number of neighbourhoods within each cluster is proportional to various neighbourhood sizes. *ADS* is applied on each cluster respectively. The effect of varying number of neighbourhoods on accuracy, privacy and execution time was analyzed and plotted in Figs. (6), (7) and (8). Figure (7) measures the relation between MAE and neighbourhood sizes, where the neighbourhood size is increased from 1 % to 50 % of its cluster size. MAE generally becomes smaller when the number of neighbourhoods is high. With small neighbourhood size, the number of neighbourhoods in each cluster is large that can better preserve the cluster distribution and reduce level of perturbation in the original dataset. The number of substituting/merging operations between these neighbourhoods has a minor effect on MAE values as these operations done within the same cluster. As for the privacy level as shown in Fig. (8), we can see that as the neighbourhood size goes smaller, the VI value generally goes low too. This is because when the granularity of clusters is finer, *ADS* will be able to extract neighbourhoods that can more accurately simulate the cluster distribution of the original dataset, thus the generated perturbated dataset can better preserves the original distribution. On the other hand, changing neighbourhood size strongly influences execution time, as substituting/merging operations on large number of neighbourhoods consumes longer time than for a small number of

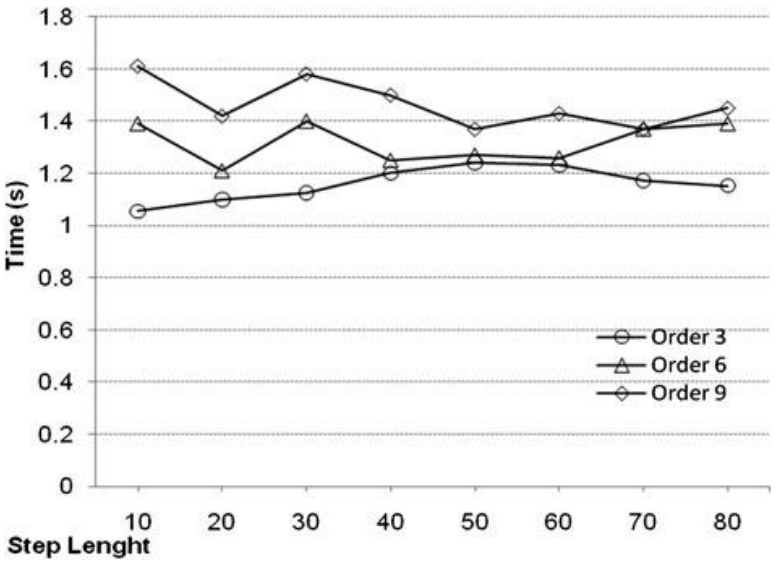


Fig. 5 Execution time for different step length in *CTA*

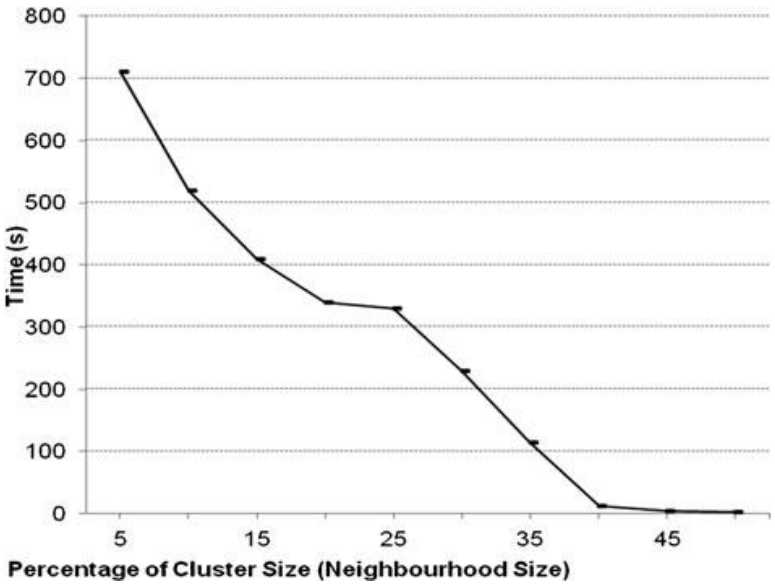


Fig. 6 Execution time for different step length in *CTA*

neighbourhoods. Figure (6) shows execution time for different neighbourhood sizes. The graph shows that, execution time decreases exponentially while increasing neighbourhood size. This result is due to reducing number of neighbourhoods that need to be processed by *ADS*. The height of extracted tree from each cluster is three, this might not be always applicable in some cases, but *ADS* will revise it in time to ensure the execution efficiency. The substitution process does not change this three level tree structure but merging can change it. Therefore, in merging procedure we need to amend elements of second tree to point to local core-point of first tree in order to maintain this three level tree structure. *ADS* algorithm does not require the whole dataset to be retained in the memory since it can be applied to limited number of clusters per time. Although the graph is exponential in nature, execution time in interval 20 %–25 % is nearly stable and it was completed in almost 327 s. This was result from equal number of merging operations on neighbourhoods in this interval. In general, the time complexity for *ADS* algorithm depends on number merging operations which is mainly depend on the number and size of neighbourhoods in the dataset. Based on these results, *ADS* is beneficial for large preferences sets collected from different participants with respect to execution time.

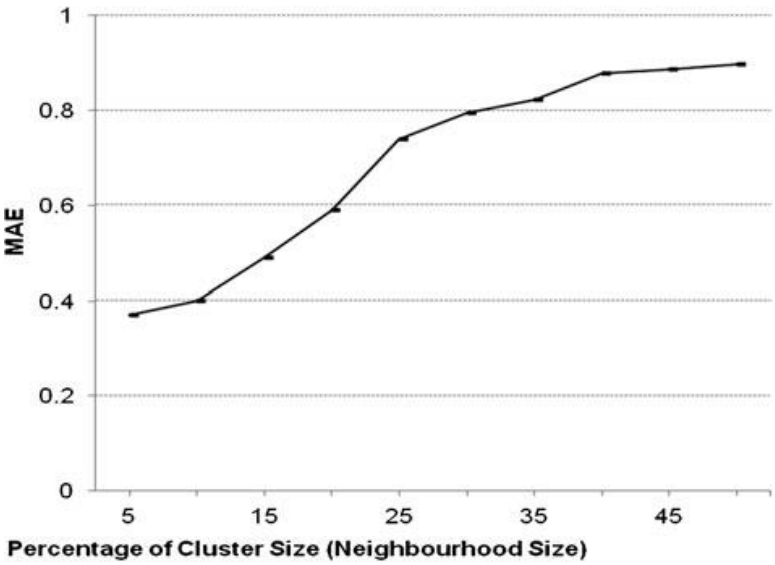


Fig. 7 Accuracy level for different neighbourhood sizes for *ADS*

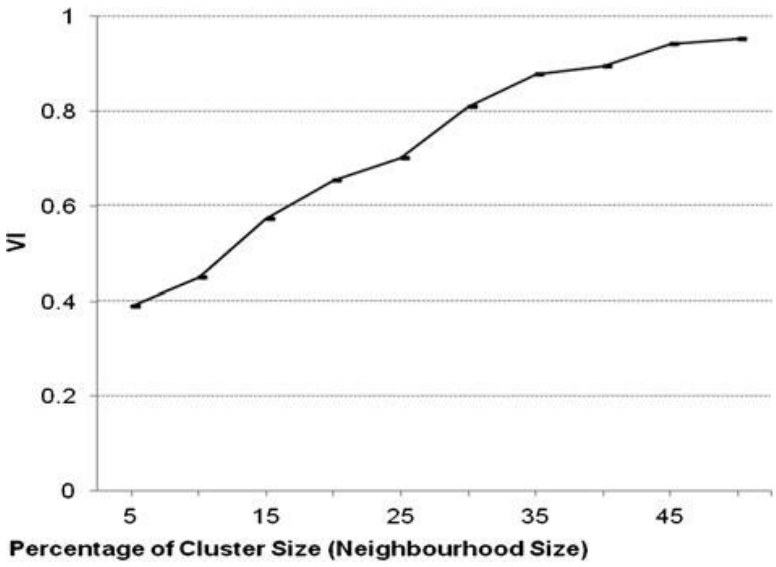


Fig. 8 Privacy level for different neighbourhood sizes for *ADS*

As we see for both of VI and MAE they generally go down with the decrease of neighbourhood size. These trends are not always like this, because in high dimensional space, the data points are generally sparse. The increase in number of neighbourhoods cannot guarantee that there are always data points to join them. So VI and MAE can sometimes augment while the neighbourhood size decreases. Generally, the MAE scales down with increase in the number of local-core points until it will be equal to the natural number of local-core points in each cluster. So it is actually a trade off between privacy, time and accuracy.

Finally, we measured the overall performance of PPS algorithm in terms of the enhancement achieved in uploading time for the collected data. Figure (9) illustrates the cumulative distribution function (CDF) of time uploading the collected data of 331,25 bytes from the participant under the PPS algorithm we proposed. PPS = $C_T \leq 34$ KB/s refers to executing PPS with a circuit throughput equal to 34 KB/s. Table 1 gives the mean, median for executing the PPS with different circuit throughput values. Our analysis of Fig. (9) and Table 1 lead us to the following observations; the performance of Tor's default path selection algorithm is unacceptable for responsive recommendations services. The largest uploading time for the data is 182.61 s; also PPS algorithm significantly improves path selection performance. With PPS = $C_T \leq 34$ KB/s, the median uploading time is 9.59 s compared with the default Algorithm 33.56 s.

12 Conclusion and future wok

In this paper, we presented our attempt to develop an agent based middleware for private recommendations service. We gave a brief overview of AMPR architecture, components and recommendation process with application to IPTV. We presented a novel two stage concealment process which provides complete privacy control to participants over their preferences. The concealment process utilizes hierarchical topology, where participants are organized into groups, from which super-peers are elected based on their reputation. One important notice, when the formed group does not contain a large enough participant base, privacy guarantees weaken. In the extreme case when there is only one participant, less privacy can be achieved when preferences are submitted. The participants need to be aware of such potential vulnerability. Super-peers & PCRS use platform for privacy preferences

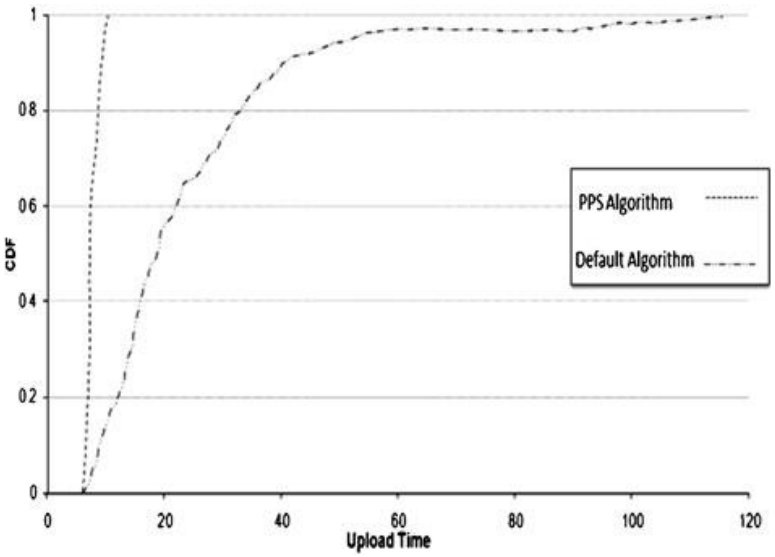


Fig. 9 Uploading time using PPS Algorithm

(P3P) policies for specifying their data usage practices. While Participants describe their privacy constraints for the data extracted from their profiles in a dynamically updateable fashion using P3P policies exchange language (APPEL). AMPR allows fine grained enforcement of privacy policies by allowing participants to ensure that the extracted preferences for specific request do not violate their privacy by automatically checking whether there is an APPEL preference corresponding to the given P3P policy. Super-peers receive preferences data obtained anonymously from underlying participants and then they conceal this data and send it to PCRS. Local obfuscation using clustering transformation algorithm (*CTA*) is used in the course of participant preferences collection, while the global perturbation using advanced data substitution algorithm (*ADS*) is used to protect the privacy of participants' preferences data collected at multiple super-peers. Dishonest behaviours are more difficult to be fully detected, as large number of submissions from malicious user is difficult to recognize as this behaviour can go undetected unless the submitted preferences significantly differs from the normal required ones. Due to that, abuses need to be detected based on other cues. In our solution, this is partially mitigated by letting participants aware of the ability to uncover their identity. We tested the performance of the proposed two stage concealment process on a real dataset. We evaluated how the overall accuracy of the recommendations depends on different privacy levels. The experimental and analysis results show that privacy increases under the proposed middleware without hampering the accuracy of the recommendation. In particular the mean absolute error can be reduced for a large data sizes with proper tuning of parameters of the proposed algorithms.

We realized that there are many challenges in building a privacy enhanced middleware for recommender services. As a result we focused on an agent based middleware scenario. A future research agenda will include utilizing game theory to better formulate user groups, sequential preferences release and its impact on privacy of whole profile. We will consider reducing transmission time and the load on the network traffic. Furthermore it is included to

Table 1 Uploading Time for Different C_T values

	default	PPS = $C_T \leq$10 KB/s	PPS = $C_T \leq$20 KB/s	PPS = $C_T \leq$30 KB/s	PPS = $C_T \leq$34 KB/s
Mean	30.38	25.34	19.64	12.54	8.41
Median	33.56	17.35	15.18	10.73	9.59

strengthen our middleware against shilling attacks, extending our scheme to be directed towards distributed collaborative filtering techniques in a P2P environment. Moreover, we need to investigate weighted features vector methods and its impact on released ratings. Such that, the participant not only obfuscates his items' ratings but also he can express specific items to be diversely obfuscated. We need to perform extensive experiments on other real datasets from the UCI repository and compare our performance with other techniques proposed in the literature. Finally we need to consider different data partitioning techniques as well as identify potential threats and add some protocols to ensure the privacy of the data against those threats.

Acknowledgments This work has received support from the Higher Education Authority in Ireland under the PRTLI Cycle 4 programme, in the FutureComm project (Serving Society: Management of Future Communications Networks and Services).

References

1. Canny J (2002) Collaborative filtering with privacy via factor analysis. Paper presented at the Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, Tampere, Finland
2. Canny J (2002) Collaborative filtering with privacy. Paper presented at the Proceedings of the 2002 IEEE Symposium on Security and Privacy
3. Carbo J, Molina J, Davila J (2002) Trust management through fuzzy reputation. Int J Cooper Inform Syst 12:135–155
4. Dingledine R, Mathewson N, Syverson P (2004) Tor: the second-generation onion router. Paper presented at the Proceedings of the 13th conference on USENIX Security Symposium - Volume 13, San Diego, CA
5. Elmisery A, Botvich D (2011) Private recommendation service ror IPTV system. In: 12th IFIP/IEEE International Symposium on Integrated Network Management, Dublin, Ireland. IEEE
6. Elmisery A, Botvich D (2011) Privacy aware recommender service for IPTV networks. In: 5th FTRA/IEEE International Conference on Multimedia and Ubiquitous Engineering, Crete, Greece. IEEE
7. Elmisery A, Botvich D (2011) Agent based middleware for maintaining user privacy in IPTV recommender services. In: 3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems. ICST, Aalborg, Denmark
8. Elmisery A, Botvich D (2011) Agent based middleware for private data mashup in IPTV recommender services. In: 16th IEEE International Workshop on Computer Aided Modeling, Analysis and Design of Communication Links and Networks. IEEE, Kyoto, Japan
9. Elmisery A, Botvich (2011) D An agent based middleware for privacy aware recommender systems in IPTV Networks. In: 3rd International Conference on Intelligent Decision Technologies University of Piraeus, Greece, KES-Springer Smart Innovations, Systems and technologies. Springer Verlag
10. Elmisery AM, Botvich D (2011) An agent based middleware for privacy aware recommender systems in IPTV networks. In: Watada J, Phillips-Wren G, Jain LC, Howlett RJ (eds) Intelligent decision technologies. Smart innovation, systems and technologies, vol 10. Springer, Berlin, Heidelberg, pp 821–832. doi:10.1007/978-3-642-22194-1_81
11. Elmisery A, Botvich D (2011) Privacy aware obfuscation middleware for mobile jukebox recommender services. In: The 11th IFIP Conference on e-Business, e-Service, e-Society, Kaunas, Lithuania, IFIP
12. Elmisery A, Botvich D (2011) Enhanced Middleware for Collaborative Privacy in IPTV Recommender Services Journal of Convergence 2 (2):10
13. Elmisery A, Huaiguo F (2010) Privacy preserving distributed learning clustering of healthcare data using cryptography protocols. In: 34th IEEE Annual International Computer Software and Applications Workshops, Seoul, South Korea
14. Esma A (2008) Experimental demonstration of a hybrid privacy-preserving recommender system. In: Gilles B, Jose MF, Flavien Serge Mani O, Zbigniew R (eds) pp 161–170
15. Fellows MR, Guo J, Komusiewicz C, Niedermeier R, Uhlmann J (2009) Graph-Based Data Clustering with Overlaps. Paper presented at the Proceedings of the 15th Annual International Conference on Computing and Combinatorics, Niagara Falls, NY

16. Gemmis Md, Iaquinta L, Lops P, Musto C, Narducci F, Semeraro G (2009) Preference Learning in recommender systems. Paper presented at the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD), Slovenia
17. Ghinita G, Kalnis P, Skiadopoulos S (2007) PRIVE: anonymous location-based queries in distributed mobile systems. Paper presented at the Proceedings of the 16th international conference on World Wide Web, Banff, Alberta, Canada
18. Gupta D, Digiovanni M, Narita H, Goldberg K (1999) Jester 2.0 (poster abstract): evaluation of an new linear time collaborative filtering algorithm. Paper presented at the Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, Berkeley, California, United States
19. Herlocker JL, Konstan JA, Terveen LG, Riedl JT (2004) Evaluating collaborative filtering recommender systems. ACM Trans Inf Syst 22(1):5–53. doi:doi.acm.org/10.1145/963770.963772
20. Huang Z, Du W, Chen B (2005) Deriving private information from randomized data. Paper presented at the Proceedings of the 2005 ACM SIGMOD international conference on Management of data, Baltimore, Maryland
21. Imani M, Taheri M, Naderi M (2010) Security enhanced routing protocol for ad hoc networks. J Conv 1 (1):43–48
22. Indyk P, Motwani R (1998) Approximate nearest neighbors: towards removing the curse of dimensionality. Paper presented at the Proceedings of the thirtieth annual ACM symposium on theory of computing, Dallas, Texas, United States
23. Jarvis RA, Patrick EA (1973) Clustering using a similarity measure based on shared near neighbors. IEEE Trans Comput 22(11):1025–1034. doi:10.1109/t-c.1973.223640
24. Kargupta H, Datta S, Wang Q, Sivakumar K (2003) On the privacy preserving properties of random data perturbation techniques. Paper presented at the Proceedings of the Third IEEE International Conference on Data Mining
25. Kelly D, Teevan J (2003) Implicit feedback for inferring user preference: a bibliography. SIGIR Forum 37 (2):18–28. doi:doi.acm.org/10.1145/959258.959260
26. Kingsford C (2009) Information theory notes
27. Klyuev V, Oleshchuk V (2011) Semantic retrieval: an approach to representing, searching and summarising text documents. Int J Inf Technol Commun Converg 1:221–234
28. Konstan J, Miller B, Maltz D, Herlocker J, Gordon L, Riedl J (1997) GroupLens: applying collaborative filtering to usenet news. Commun ACM 40(3):77–87. doi:citeulike-article-id:486168
29. Liu K, Giannella C, Kargupta H (2006) An attacker's view of distance preserving maps for privacy preserving data mining knowledge discovery in databases: PKDD 2006. In: Fürnkranz J, Scheffer T, Spiliopoulou M (eds) Lecture notes in computer science vol 4213. Springer, Berlin / Heidelberg, pp 297–308. doi:10.1007/11871637_30
30. McSherry F, Mironov I (2009) Differentially private recommender systems: building privacy into the net. Paper presented at the Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Paris, France
31. Miller BN, Konstan JA, Riedl J (2004) PocketLens: toward a personal recommender system. ACM Trans Inf Syst 22(3):437–476. doi:doi.acm.org/10.1145/1010614.1010618
32. Nejdl W, Wolpers M, Siberski W, Schmitz C, Schlosser M, Brunkhorst I, L\ A, \#246, ser (2003) Super-peer-based routing and clustering strategies for RDF-based peer-to-peer networks. Paper presented at the Proceedings of the 12th international conference on World Wide Web, Budapest, Hungary
33. Pingley A, Yu W, Zhang N, Fu X, Zhao W (2009) CAP: a context-aware privacy protection system for location-based services. Paper presented at the Proceedings of the 2009 29th IEEE International Conference on Distributed Computing Systems
34. Polat H, Du W (2003) Privacy-preserving collaborative filtering using randomized perturbation techniques. Paper presented at the Proceedings of the Third IEEE International Conference on Data Mining
35. Polat H, Du W (2005) SVD-based collaborative filtering with privacy. Paper presented at the Proceedings of the 2005 ACM symposium on Applied computing, Santa Fe, New Mexico
36. Pyshkin E, Kuznetsov A (2010) Approaches for web search user interfaces: how to improve the search quality for various types of information. Journal of Convergence 1:1–8
37. Reaz A, Raouf B (2010) A scalable peer-to-peer protocol enabling efficient and flexible search
38. Sweeney L (2002) k-anonymity: a model for protecting privacy. Int J Uncertain Fuzziness Knowl-Based Syst 10 (5):557–570. doi:10.1142/s0218488502001648
39. Ye Y, Li X, Wu B, Li Y (2010) A comparative study of feature weighting methods for document co-clustering. Int J Inf Technol Commun Converg 1(2):206–220

Appendix C: Jukebox Recommender Service Scenario

Article IX

Privacy Aware Obfuscation Middleware for Mobile Jukebox Recommender Services

Ahmed M. Elmisery, Dmitri Botvich

In Proceedings of the 11th IFIP WG 6.11 Conference on e-Business, e-Services, and e-Society, (I3E 2011), Kaunas, Lithuania, October 2011.

Privacy Aware Obfuscation Middleware for Mobile Jukebox Recommender Services

Ahmed M. Elmisery and Dmitri Botvich

Waterford Institute of Technology – WIT
Telecommunications Software and Systems Group – TSSG, Co. Waterford, Ireland
{ael-misery, dbotvich}@tssg.org

Abstract. Mobile Jukebox is a service offered by mobile operators to their clients, such that subscribers can buy or download anywhere, anytime full-length music tracks over the 3G Mobile networks. Unlike some music download services, the subscribers can reuse the selected tracks on their music players or computers. As the amount of online music grows rapidly, Jukebox providers employ automatic recommender service as an important tool for music listeners to find music that they will appreciate. On one hand, Jukebox recommender service recommend music based on users' musical tastes and listening habits which reduces the browsing time for searching new songs and album releases. On the other hand, users care about the privacy of their preferences and individuals' behaviors regarding the usage of recommender service. This work presents our efforts to design an agent based middle-ware that enables the end-user to use Jukebox recommender services without revealing his sensitive profile information to that service or any third party involved in this process. Our solution relies on a distributed multi-agent architecture involving local agents running on the end-user mobile phone and two stage obfuscation process used to conceal the local profiles of end-users with similar preferences. The first stage is done locally at the end user side but the second stage is done at remote nodes that can be donated by multiple non-colluding end users that requested the recommendations or third parties mash-up service. All the communications between participants are done through anonymised network to hide their network identity. In this paper, we also provide a mobile jukebox network scenario and experimentation results

Keywords: Privacy, Clustering, Mobile Jukebox, Recommender Service, Multi-Agent.

1 Introduction

Being music a very important thing in people's life, music applications are present in every computer and over many mobile devices. For such reason, mobile operators offer Mobile Jukebox as a moderate price service to their clients, such that subscribers can buy or download full-length music tracks over 3G Mobile networks. Unlike some music download services, the user can reuse the selected tracks on their music players or portable music devices.

According to [1] the more services appear in future, the more demand for personalization services will be to fight against information overload and find information relevant to each user. Recommender services can be seen as a suitable solution to these problems as they customize the offered services according to unique and individual needs of each user. The Jukebox providers employ recommender services to reduce browsing time for music listeners as the amount of online music grows rapidly. Jukebox recommender service becomes an increasingly important tool for music listeners to easily find new songs or playlists that they will appreciate. Examples of available Jukebox services are Apple iTunes® and Last.fm®. Apple iTunes automatically generates a playlist of songs from the user's library which is similar to the selected songs, While Last.fm builds a detailed profile for each user's musical taste by recording details of the songs the user listens to either from internet radio stations or user's computer or portable music devices. This information is transferred to Last.fm's database via music player and the user's profile data can be displayed on his profile page.

Jukebox recommenders commonly use collaborative filtering (CF) techniques to recommend music based on the listening behaviors of other music listeners. The Jukebox recommender harness the "wisdom of the crowds" to recommend music. Even though they generate good recommendations there are still some problems like the cold-start problem, as a recommender needs a significant amount of data before it can generate appropriate recommendations. However the acquisition, storage and application of sensitive personal information cause privacy concerns for users. There are many things having an impact on the perception of privacy for users like what kind of information is collected, how the information is used and the degree of accessibility of the information by others.

The authors in [1] have done an empirical research concerning privacy preferences and individuals' behaviors regarding personalization in music recommender systems. They found out that information about the purpose of the disclosure, recipients of the information, and the degree of the information involved and the benefits users expect to gain from disclosing personal information are the main factors influencing disclosure behavior. Based on their questionnaire in [2], participants were more willing to disclose music preferences than their personality. Participants considered information about personality traits more personal and more sensitive information than preferences for music genres. Participants expressed worries about not knowing how their information will be used in the system and who gets access to their personal information. The sensitivity of information affects on the disclosure decision. The questionnaire also shows that some participants even consider what benefits they will gain from disclosing the information. Participants can be divided into two groups based on their disclosure behavior, depending on whether they want to disclose anonymously or including identity information. One important factor have an impact on people's disclosure behavior is the security and privacy standards taken by the Jukebox providers.

In this work, we proceed with our approach presented in [3-7] to build AMPR (i.e. acronym for agent based middleware for private recommendations) that allows end-

users to receive useful recommendations without disclosing their real preferences to the service. In the following section we will describe some properties for AMPR:

1. AMPR is running as a multi-agent based middleware to support different types of clients either thin or thick. Moreover, this architecture enables smooth integration with wide range of existing recommender services
2. AMPR preserves the aggregates in the obfuscated profiles to maximize their utility in order to attain acceptable recommendations accuracy, which facilitate AMPR to work with different state-of art filtering algorithms. Extra overhead in computation and communication to be added in the recommendation process due to the two stage obfuscation process
3. AMPR employs two stage obfuscation process to conceal the user's preferences in his profile. The real user profile doesn't leave his mobile phone until it is properly desensitized and it is maintained encrypted with private password that is known only to the user. If the user doesn't accept to be tracked by the recommender service using his network identity, AMPR hides his identity by routing the submission of his locally obfuscated profile through relaying nodes in an anonymous communication network before sending it to the recommender service.

In the rest of this paper we will generically refer to songs and playlists as Items. In section 2, we describe some related work. Section 3 introduces recommender system for mobile jukebox service scenario landing AMPR. Section 4 introduces our proposed solution. Section 5 describes the recommendation strategy used in our PCRS. Section 6 presents some experiments and results based on our proposed solution. Section 7 includes conclusions and future work.

2 Related Work

The majority of the literature addresses the problem of privacy for recommender services based on collaborative filtering technique, Due to it is a potential source of leakage of private information shared by the users as shown in [8]. In [9] it is proposed a theoretical framework to preserve the privacy of customers and the commercial interests of merchants. Their system is a hybrid recommender that uses secure two party protocols and public key infrastructure to achieve the desired goals. In [10, 11] it is proposed a privacy preserving approach based on peer to peer techniques using users' communities, where the community will have a aggregate user profile representing the group as whole and not individual users. Personal information will be encrypted and the communication will be between individual users and not servers. Thus, the recommendations will be generated at client side. In [12, 13] it is suggest another method for privacy preserving on centralized recommender systems by adding uncertainty to the data by using a randomized perturbation technique while attempting to make sure that necessary statistical aggregates such as mean don't get disturbed much. Hence, the server has no knowledge about true values of individual rating profiles for each user. They demonstrate that this method does not decrease essentially the obtained accuracy of the results. Recent research work [14, 15] pointed

out that these techniques don't provide levels of privacy as it was previously thought. In [15] it is pointed out that arbitrary randomization is not safe because it is easy to breach the privacy protection it offers. They proposed a random matrix based spectral filtering techniques to recover the original data from perturbed data. Their experiments revealed that in many cases random perturbation techniques preserve very little privacy. Similar limitations were detailed in [14].

3 Recommender System for Mobile Jukebox Service - Scenario

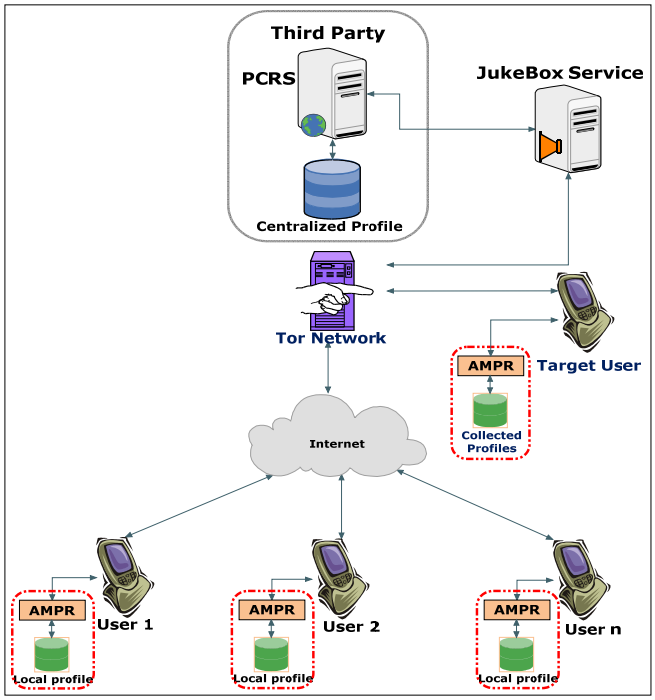


Fig. 1. Jukebox Service for Mobile Users with Third Party Private Recommender Service

We consider the scenario where a private centralized recommender service (PCRS) is implemented on an external third party server and end-users give information about their preferences to that server in order to receive music recommendations. The user preferences stored in his profile in the form of ratings or votes for different item, such that items are rated explicitly or implicitly on a scale from 1 to 5. An item with rating of 1 indicates that the user dislikes it while a rating of 5 means that the user likes it. PCRS collects and stores different users' preferences in order to generate useful recommendations.

In this scenario there are two possible ways for user's discloser: through his personal preferences included in his profile [16] or through the user's network address

(IP). AMPR employs two principles to eliminate these two disclosure channels, respectively. The obfuscation agents perturb user's preferences for different items in his profile and the synchronize agent hides the user's network identity by routing the communication with other participants through relaying nodes in Tor [17] anonymous network. The main challenge for synchronize agent is to tune up Tor and optimize its performance while maintaining the user anonymity.

We don't assume the server to be completely malicious. This is a realistic assumption because the service provider needs to accomplish some business goals and increase its revenues. In our framework, we will use the mobile phone storage to store the user profile. On the other hand, the Jukebox service maintains a centralized rating database that is used by the PCRS; Figure (1) shows the architecture of our approach. Additionally, we alleviate the user's identity problems stated above by using anonymous pseudonyms identities for users.

4 Proposed Solution

In the next sub-sections, we will present our proposed middleware for protecting the privacy of users' profiles

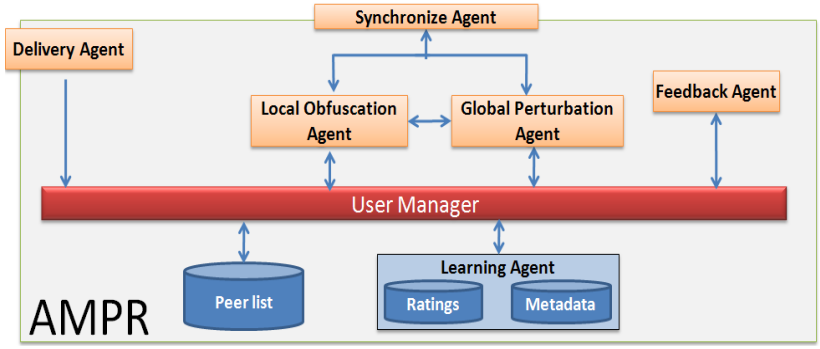


Fig. 2. AMPR Components

Figure (2) demonstrates AMPR components that are running in the mobile phone at the user side. As shown, AMPR consists of different co-operative agents, A Learning agent captures user preferences about items explicitly or implicitly to build a rating table and meta-data table. The local obfuscation agent implements *CTA* obfuscation algorithm to achieve user privacy while sharing the data with other users or the system. The global perturbation agent is only invoked if the user is acting as a target user in recommendation process; it executes *EVS*-algorithm on the collected profiles finally. The synchronize agent is responsible for selecting the best suitable routing paths in the anonymised network to enhance its performance.

The recommendation process based on the two stage obfuscation process in our framework can be summarized as following:

1. The learning agent collects user's preferences about different items which represent a local profile. The local profile is stored in two databases, the first one is the rating

database that contains (item_id, rating) and the other one is the metadata database that contains the feature vector for each item (item_id, feature1, feature2, feature3). The feature vector can include genre, author, album, decade, vocalness, singer, instruments, number of reproductions and so on.

2. The target user broadcast a message to other users near him to request recommendations for specific genre or category of items. Individual users who decide to respond to that request use their local obfuscation agent to obfuscate a part of their local profiles that match query. The group members submit their locally obfuscated profiles to the requester using an anonymised network like TOR to hide their network identities. Enhancing the performance of communication through Tor is discussed in the next sub-section. If the size of group formation less than a specific value, the target user contacts the PCRS directly to gets recommendation from the centralized profiles stored in it.
3. In order to hide items identifiers and meta-data from the requester and the PCRS. The manger agent at each participant side use locality-sensitive hashing (LSH) [18] to hash these values. One interesting property for LSH is that similar items will be hashed to the same value with high probability. PCRS is still be able to perform computations on the hashed items using appropriate distance metrics like hamming distance or dice coefficient.
4. After the target user receives all the participants' profiles (group profile), he/she incites his global perturbation agent to perturb the collected profiles. Then he can interact with PCRS by acting as an end-user has group profile as his own profile. The target user submits his/her group profile through an anonymised network to PCRS in order to attain recommendations.
5. PCRS performs its filtering techniques on the group profile which in turn return a list of items that are correlated with that profile. This list is encrypted with a private key provided by target-user and it is sent back on the reverse path to the target user that in turn decrypts and publishes it anonymously to the other users that participated in the recommendation process.

4.1 Enhancing The Anonymized Network (Tor) Performance

Tor [17] is an anonymity network based on the original onion routing design but with several modifications in terms of security, efficiency, and deployability. The Tor network includes a small set of trusted authoritative directory servers responsible for aggregating and distributing signed information about known routers in the network. Tor clients periodically fetch the directory information from directory mirrors in order to learn information about other servers in the network.

Tor network suffers from serious performance degradation because of its random path selection algorithms that use self-reported bandwidth values only, which might select with high probability a router with low bandwidth because it is sensitive to loads and changing network conditions. The synchronize agent seeks to enhance the performance by partitioning the Tor network into classes of high or low bandwidth Tor routers to better understand the relationships between different classes of routers /potential paths. Paths drawn from the class of high-bandwidth routers can provide better performance. Paths can be reserved for participating in specific

recommendation requests based on a user's priorities or preferences. Therefore, inspired from the work in [19] we have implemented a simple parameterized path selection algorithm (*PPS*) that allow the synchronize agent to enhance the path selection in the Tor network with two priorities and it can be easily extended to support priorities larger than two. The synchronize agent can create circuits in advance to reduce the waiting time then measure the path throughput P_T before using each path. *PPS* consists of the following steps:

1. The user input minimum path throughput P_T, and circuit throughput C_T to the synchronize agent.
2. Based on Tor authoritative directory servers, the algorithm start partitioning the class of high-bandwidth routers into a set of overlapping clusters (based on geographical location, platform, bandwidth, uptime, last update, number of connections and self-reported bandwidth estimate) using the algorithm on [20].
3. It builds a pool of Tor nodes whose bandwidth $\geq P_T$ from each cluster. In order to decrease the delay in circuit creation, the synchronize agent can select overlapping routers between clusters
4. Then, it randomly builds circuits passing through these clusters. Then measure each circuit throughput, and select the first circuit that achieve bandwidth $\geq C_T$.
5. The synchronize agent then negotiates session keys with each router in the circuit. The exit router is responsible for establishing the connection from the Tor network to the client's intended destination
6. The synchronize agent records the previously used Tor nodes and exclude them from future circuit building clusters.

4.2 Proposed obfuscation Algorithms

In the next subsections, we present two different algorithms used by the obfuscation agents in AMPR to obfuscate the user profile in a way that secure user's preferences in PCRS with minimum loss of accuracy.

Local Obfuscation using Clustering Transformation Algorithm (*CTA*).

We proposed a novel algorithm for obfuscating the user profile before sharing it with other users. *CTA* designed especially for the sparse data problem we have here. *CTA* partitions the user profile into smaller clusters and then pre-process each cluster such that the distances inside the same cluster will maintained in its obfuscated version. We use local learning analysis (*LLA*) clustering method proposed in [21] to partition the dataset. After complete the partitioning, we embed each cluster into a random dimension space so the sensitive ratings will be protected. Then the resulting cluster will be rotated randomly. In such a way, *CTA* obfuscates the data inside user profile while preserving the distances between the data points to provide high accurate results when performing recommendations. The algorithm consists of the following steps:

1. The user ratings is stored in his mobile phone as dataset D consists of c rows, where each row is a sequence of X attributes where $X = x_1\ x_2\ x_3 \dots \dots \dots\ x_n$.

2. The dataset D is portioned vertically into $D_1\ D_2\ D_2 \dots \dots D_m$ subsets of length L, if n/L is not perfectly divisible then *CTA* randomly selects attributes already assigned to any subset and joins them to the attributes of the incomplete subsets.
3. Cluster each subset $\forall_{j=1}^{m} D_j$ Using *LLA* algorithm, that result in K clusters $D_j = C_{j1}, C_{j2}, C_{j3} \dots . C_{jk}$ for each subset. So every point in the original dataset D falls exactly in one cluster. The aim of this step is to increase the privacy level of the transformation process and make reconstruction attacks difficult.
4. *CTA* generates two sets for each cluster in the subset D_j these are H_{ji} and O_{ji}. Where H_{ji} is the set of points with highest values for field function and O_{ji} is the rest of points in C_{ji}. For each point $x_{1i} \in H_{ji}$ construct a weighted graph Γ_i that contains its k-nearest neighbours in O_{ji}, each edge $e \in \Gamma_i$ has a weight equals to the influence function of that point $f_{Gauss}^{b_{ji}}(x_{1i})$.
5. Estimate the geodesic distances by Computing the shortest distance between each two points in graph Γ_i using Dijkstra or Floyd algorithm and then build a distance matrix $D_{\Gamma_i} = \{ f_{Gauss}^{b_i}(x_i) \}$.
6. Based on D_{Γ_i}, we find a *d*-dim embedding space C'_{ji} using classical *MDS* [22] as follows
   - Calculate the matrix of squared distances $S = D_{\Gamma_i}^2$ and the centering matrix $H = 1 - 1/N\, ee^T$
   - The characteristic vectors are chosen to minimize $E = \left\| \tau(D_{\Gamma_i}) - \tau(D_d) \right\|_{L^2}$, where $\tau(D_d)$ is the distance matrix for the *d-dim* embedding space, and converts distances to inner products $\tau = -HSH/2$.
7. For each cluster $\forall_{j=1}^{m} \forall_{i=1}^{k} C'_{ji}$, *CTA* randomly select two attributes x_a and x_b to perform rotation perturbation on selected attributes $R(x_a, x_b)$ using transformation matrix M_j^{θ} setup by the user for each cluster using range of angles defined in advance by the user.
8. Repeating steps 4-7 for all clusters in $\forall_{j=1}^{m} D_j$ to get the obfuscated portion D'_j. Finally, the obfuscated dataset is obtained by $D' = \cup_{j=1}^{n} D'_j$.

Global Perturbation using Enhanced Value-Substitution (EVS) Algorithm.
After executing the local obfuscation process the global perturbation phase starts. The key idea for *EVS* is based on the work in [23] that uses Hilbert curve to maintain the association between different dimensions. In this subsection, we extend this idea as following, we also use Hilbert curve to map m-dimensional profile to 1-dimensional profile then *EVS* discovers the distribution of that1-dimensional profile. Finally, we perform perturbation based on that distribution in such a way to preserve the profile range. The steps for *EVS* algorithm consists of the following steps:

1. We denote the collected m-dimensional user profiles as dataset D of c rows, where each row is a sequence of m dimensions $A = A_1, A_2, A_3, A_4 \dots \dots, A_m$.
2. *EVS* divides the *m*-dimensional profile into grids of order k (where k is user defined value) as shown in [23, 24]. For order k, the range for each dimension divided into 2^k intervals.

3. For each dimension $\forall_{i=1}^{m} A_i$ of the collected profile :
   - Compute the *k*-order Hilbert value for each data point $\forall_{x=1}^{c} a_{ix}$. This value represents the index of the corresponding interval where it falls in.
   - *EVS* sort the Hilbert values from smallest to biggest, then use the step length (a user defined parameter) to measure whether any two values are near from each other or not. If these values are near, they are placed in the same partition $\forall_{v=1}^{k} k_{iv}$.

   These two steps iterates for all *m*-dimensions. The final result from these steps is partitions for each dimension denoted as $\forall_{i=1}^{m} \forall_{v=1}^{k} C_{iv}$
4. EVS constructs a *N* shared nearest neighbour sets S_r where $r = 1 \ldots . N$ as in [25] from different partitions with a new modified similarity function as following, two partitions in different dimensions C_{iv} , C_{i+1v} form a shared nearest neighbour set S_r if they share *k*-number of common elements such that $S_r = C_{iv} \cup C_{i+1v}$
5. For each newly created set S_r , EVS calculates the interquartile range. Then, for each point $a_i \in S_r$ generate a uniform distributed random point n in that range that can substitutes a_i.
6. Finally, the new set $D' = \cup_{r=1}^{N} S_r$ is sent to PCRS

5 Recommendation Strategy

PCRS employ online mode filtering algorithms to make predictions on the ratings of a particular user by collecting preference information from other users. The collected profiles (group profile) represented as $m * n$ user item matrix which contains a collection of numerical obfuscated ratings of *M* users on *N* items. After that, the neighbourhood formation at PCRS is done by calculating the similarity between users in the user-item matrix. Users similar to the target user using some proximity metric will form a proximity based neighbourhood with him [12]. This neighbourhood will utilize later for predication step. The prediction on rating of user i for item *K* is given by a weighted average [26] of users whose ratings are similar to the target user.

$$P_{ik} = \overline{v_i} + \frac{\sum_{j \in U_k} s(u_i, u_j)(v_{jk} - \overline{v_j})}{\sum_{j \in U_k} |s(u_i, u_j)|}$$

Where $U_k = \{i \in U | v_{ik} \neq \emptyset\}$ is the set of users who have rated the $k-$th item. $\overline{v_j}$ is the mean of all ratings made by user i. The weights of average $s(u_i, u_j)$ are the similarity between user u_i and u_j such as the Pearson correlation coefficient or Euclidean distance. We represent the user as a vector consists of *n* features slots, one for each item. These slots contain user's ratings for different items or $\emptyset$. The similarity between users' vectors is calculated as the cosine of the angle formed between them as following:

$$s(u_i, u_j) = \frac{\sum_{k=1}^{n} v_{ik} v_{jk}}{\sqrt{v_{i1}^2 + \cdots + v_{in}^2}\sqrt{v_{j1}^2 + \cdots + v_{jn}^2}}$$

Finally, the recommendation process is to produce a predicted rating based on the neighbourhood for a list of items that have not been rated by the user, these items have a high potential of interest (predicated with high positive rating) to the user. The resulting items list can be further constrained based on marketing or Qos rules.

6 Experiments

The proposed algorithms are implemented in C++. We used message passing interface (MPI) for a distributed memory implementation of *EVS* algorithm to mimic a distributed reliable network of peers. We evaluated the proposed algorithms from two different aspects: privacy achieved and accuracy of results. The experiments presented here were conducted using the Movielens dataset provided by Grouplens [27]. The dataset contains users' ratings on movies using discrete value between 1 and 5. The data in our experiments consists of 100.000 ratings for 1.682 items by 943 users. The experiments involve dividing the data set into a training set and testing set. The training set is obfuscated then used as a database for the PCRS. Each rating record in the testing set is divided into rated items and unrated items. The rated items are presented to the PCRS for making predication for the unrated items. To evaluate the accuracy of generated predications, we used the mean absolute error (*MAE*) metric proposed in [28]. *MAE* measures the predication verity between the predicated ratings and the real ratings, so smaller *MAE* means better recommendations provided by PCRS. To measure the privacy or distortion level achieved using our algorithms, we use variation of information metric *VI* [29] to estimate data error. Where, the higher *VI* means the larger distortion between the obfuscated and original dataset, which means higher privacy level.

To evaluate the accuracy of *CTA* algorithm with respect to different number of dimensions in user profile, we control *d-dim* parameters of *CTA* to vary number of dimensions during the evaluation. Figure (3) shows the performance of recommendations of locally obfuscated data, as shown the accuracy of recommendations based on obfuscated data is little bit low when *d-dim* is low. But at a certain number of dimensions (500), the accuracy of recommendations of obfuscated data is nearly equal to the accuracy obtained using original data.

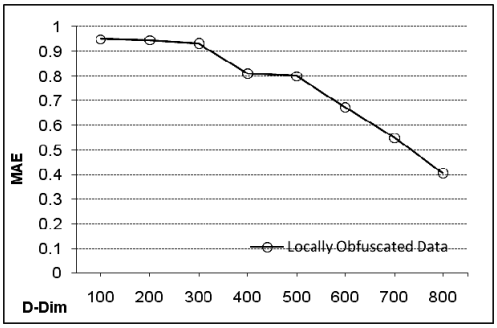


Fig. 3. Accuracy of recommendations for obfuscated dataset using *CTA*

In the second experiment performed on *CTA* algorithm, we examine the effect of *d-dim* on *VI* values. As shown in Figure (4), *VI* values decrease with respect to the increase in *d-dim* values in user profile. *d-dim* is the key element for privacy level where smaller *d-dim* value, the higher *VI* values (privacy level) of *CTA*. However, clearly the highest privacy is at *d-dim*=100. There is a noticeable drop of *VI* values

when we change *d-dim* from 300 to 600.*d-dim* value 400 is considered as a critical point for the privacy.Note that rotation transformation adds extra privacy layer to the data and in the same time maintains the distance between data points to enable PCRS to build accurate recommendation models.

In the first experiment performed on *EVS* algorithm, we measured the relation between different Hilbert curve parameters (order and step length) on the accuracy and privacy levels attained. We map the locally obfuscated dataset to Hilbert values using order 3, 6 and 9. We gradually increased the step length from 10 to 80. Figure (5) shows the accuracy of recommendations based on different step length and curve order. We can see that as the order increases, the obfuscated data can offer better predictions for the ratings. This is because as the order has higher value, the granularity of the Hilbert curve becomes finer. So, the mapped values can preserve the data distribution of the original dataset. On the other hand, selecting larger step length increases *MAE* values as large partitions are formed with higher range to generate random values from it, such that these random values substitute real values in the dataset.

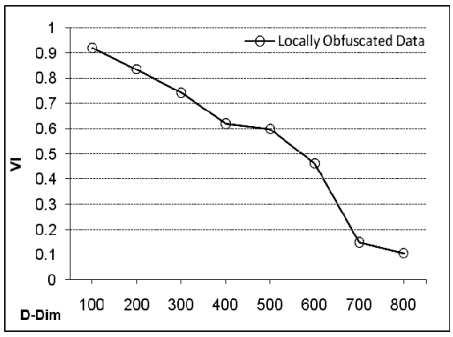


Fig. 4. Privacy levels for the obfucated dataset using *CTA*

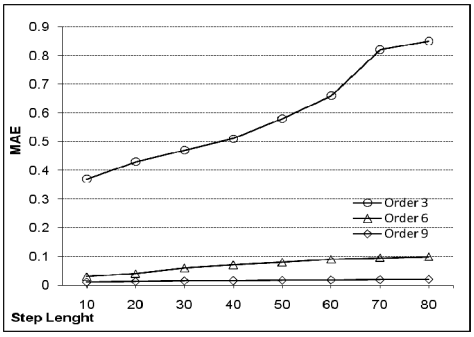


Fig. 5. Accuracy level for different step length and orders for *EVS*

As shown in Figure (6), when the order increases a smaller range is calculated within each partition which introduces less substituted values compared with lower orders that attain higher *VI* values. The reason for this is larger order divides the *m*-dimensional profile into more grids, which makes Hilbert curve better reflects the data distribution. Also, we can see that for the same Hilbert curve order the *VI* values are generally the same for different step length except for order 3, in which *VI* values has a sharp increase when step length grows from 50 to 60.The effect of increasing step length on *VI* values is more sensible in lower curve orders as fewer girds are formed and the increase of step length covers more portions of them, which will introduce a higher range to generate random values from it. So the target user should select *EVS* parameters in such a way to achieve a trade off between privacy and accuracy.

Finally, we measured the overall performance of *PPS* algorithm in terms of the enhancement achieved in uploading time for the collected profiles. Figure (7) illustrates the cumulative distribution function (*CDF*) of time uploading the collected

profiles of 331,25 bytes from the target user under the proposed *PPS* algorithm. The term *PPS* = $C_T \leq$34KB/s refers to executing *PPS* with a circuit throughput equal to 34KB/s. Table (1) gives the mean, median for execution of *PPS* with different circuit throughput values. Our analysis of Figure (7) and Table (1) lead us to the following observations; the performance of Tor's default path selection algorithm is unacceptable for responsive recommender services. The largest uploading time for the profiles is 182.61s; also our *PPS* algorithm significantly improves path selection performance.

	default	PPS= $C_T \leq$10 KB/s	PPS= $C_T \leq$20 KB/s	PPS= $C_T \leq$30 KB/s	PPS= $C_T \leq$34 KB/s
Mean	30.38	25.34	19.64	12.54	8.41
Median	33.56	17.35	15.18	10.73	9.59

Table 1: Uploading Time for Different C_T values

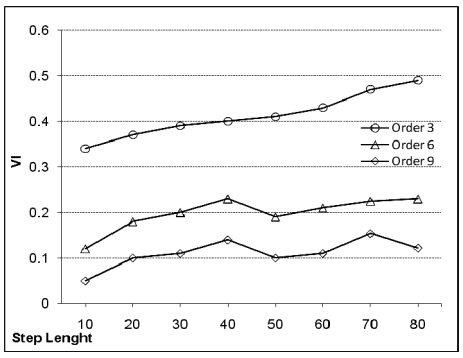


Fig. 6. Privacy level for different step length and orders for *EVS*

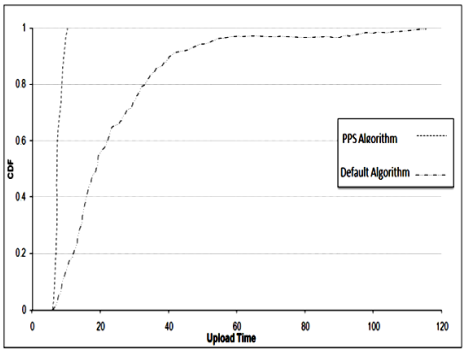


Fig. 7. Uploading time using PPS Algorithm

7 CONCLUSION AND FUTURE WOK

In this paper, we presented our ongoing work on building an agent based middleware for private recommendation services. We gave a brief overview of the recommendations process with application to Jukebox music recommendations. Also we presented the novel algorithms that provide the users with complete control of the privacy of their profiles using two stage obfuscation process. We tested The performance of these proposed algorithms on real dataset. The experiential results show that preserving users' privacy for Jukebox recommender service is possible. In particular mean average error can be reduced with proper tuning of the algorithms' parameters for large number of users. We realized that there are many challenges in building an agent based middleware scenario. This allow us to move forward in building an integrated system while studying issues such as a dynamic data release at a

later stage and deferring certain issues such as virtualized schema and auditing to future research agenda. We need to perform extensive experiments in other real data set from the UCI repository and compare the performance with other techniques. Also we need to consider different data partitioning techniques as well as identify potential threats and add some protocols to ensure the privacy of the data against those threats.

Acknowledgments : This work has received support from the Higher Education Authority in Ireland under the PRTLI Cycle 4 programme, in the FutureComm project (Serving Society: Management of Future Communications Networks and Services).

References

1. Perik, E., de Ruyter, B., Markopoulos, P., Eggen, B.: The Sensitivities of User Profile Information in Music Recommender Systems. Proceedings of Private, Security, Trust (2004)
2. Perik, E., de Ruyter, B., Markopoulos, P.: Privacy & Personalization: Preliminary Results of an Empirical Study of Disclosure Behavior. Proceedings of PEP, Edinburgh, UK. (2005)
3. Elmisery, A., Botvich, D.: An Agent Based Middleware for Privacy Aware Recommender Systems in IPTV Networks. 3rd International Conference on Intelligent Decision Technologies Springer Verlag, University of Piraeus, Greece (2011)
4. Elmisery, A., Botvich, D.: Agent Based Middleware for Private Data Mashup in IPTV Recommender Services. 16th IEEE International Workshop on Computer Aided Modeling, Analysis and Design of Communication Links and Networks. IEEE, Kyoto, Japan (2011)
5. Elmisery, A., Botvich, D.: Agent Based Middleware for Maintaining User Privacy in IPTV Recommender Services. 3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems. ICST, Aalborg, Denmark (2011)
6. Elmisery, A., Botvich, D.: Privacy Aware Recommender Service for IPTV Networks. 5th FTRA/IEEE International Conference on Multimedia and Ubiquitous Engineering. IEEE, Crete, Greece (2011)
7. Elmisery, A., Botvich, D.: Private Recommendation Service For IPTV System. 12th IFIP/IEEE International Symposium on Integrated Network Management. IEEE, Dublin, Ireland (2011)
8. McSherry, F., Mironov, I.: Differentially private recommender systems: building privacy into the net. Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, Paris, France (2009) 627-636
9. Esma, A.: Experimental Demonstration of a Hybrid Privacy-Preserving Recommender System. In: Gilles, B., Jose, M.F., Flavien Serge Mani, O., Zbigniew, R. (eds.), Vol. 0 (2008) 161-170
10. Canny, J.: Collaborative filtering with privacy via factor analysis. Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, Tampere, Finland (2002) 238-245
11. Canny, J.: Collaborative Filtering with Privacy. Proceedings of the 2002 IEEE Symposium on Security and Privacy. IEEE Computer Society (2002) 45
12. Polat, H., Du, W.: Privacy-Preserving Collaborative Filtering Using Randomized Perturbation Techniques. Proceedings of the Third IEEE International Conference on Data Mining. IEEE Computer Society (2003) 625
13. Polat, H., Du, W.: SVD-based collaborative filtering with privacy. Proceedings of the 2005 ACM symposium on Applied computing. ACM, Santa Fe, New Mexico (2005) 791-795

14. Huang, Z., Du, W., Chen, B.: Deriving private information from randomized data. Proceedings of the 2005 ACM SIGMOD international conference on Management of data. ACM, Baltimore, Maryland (2005) 37-48
15. Kargupta, H., Datta, S., Wang, Q., Sivakumar, K.: On the Privacy Preserving Properties of Random Data Perturbation Techniques. Proceedings of the Third IEEE International Conference on Data Mining. IEEE Computer Society (2003) 99
16. Parameswaran, R., Blough, D.M.: Privacy preserving data obfuscation for inherently clustered data. Int. J. Inf. Comput. Secur. **2** (2008) 4-26
17. Dingledine, R., Mathewson, N., Syverson, P.: Tor: the second-generation onion router. Proceedings of the 13th conference on USENIX Security Symposium - Volume 13. USENIX Association, San Diego, CA (2004) 21-21
18. Indyk, P., Motwani, R.: Approximate nearest neighbors: towards removing the curse of dimensionality. Proceedings of the thirtieth annual ACM symposium on Theory of computing. ACM, Dallas, Texas, United States (1998) 604-613
19. Pingley, A., Yu, W., Zhang, N., Fu, X., Zhao, W.: CAP: A Context-Aware Privacy Protection System for Location-Based Services. Proceedings of the 2009 29th IEEE International Conference on Distributed Computing Systems. IEEE Computer Society (2009) 49-57
20. Fellows, M.R., Guo, J., Komusiewicz, C., Niedermeier, R., Uhlmann, J.: Graph-Based Data Clustering with Overlaps. Proceedings of the 15th Annual International Conference on Computing and Combinatorics. Springer-Verlag, Niagara Falls, NY (2009) 516-526
21. Elmisery, A., Huaiguo, F.: Privacy Preserving Distributed Learning Clustering Of HealthCare Data Using Cryptography Protocols. 34th IEEE Annual International Computer Software and Applications Workshops, Seoul, South Korea (2010)
22. Borg, I., Groenen, P.J.F.: Modern Multidimensional Scaling: Theory and Applications (Springer Series in Statistics). Springer (2005)
23. Ghinita, G., Kalnis, P., Skiadopoulos, S.: PRIVE: anonymous location-based queries in distributed mobile systems. Proceedings of the 16th international conference on World Wide Web. ACM, Banff, Alberta, Canada (2007) 371-380
24. Reaz, A., Raouf, B.: A Scalable Peer-to-peer Protocol Enabling Efficient and Flexible Search. (2010)
25. Jarvis, R.A., Patrick, E.A.: Clustering Using a Similarity Measure Based on Shared Near Neighbors. IEEE Trans. Comput. **22** (1973) 1025-1034
26. Konstan, J., Miller, B., Maltz, D., Herlocker, J., Gordon, L., Riedl, J.: GroupLens: Applying Collaborative Filtering to {Usenet} News. Communications of the ACM **40** (1997) 77-87
27. Lam, S., Herlocker, J.: MovieLens Data Sets. Department of Computer Science and Engineering at the University of Minnesota. (2006)
28. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. **22** (2004) 5-53
29. Kingsford, C.: Information Theory Notes. (2009)

Appendix C: Jukebox Recommender Service Scenario

Article X

Holistic Collaborative Privacy Framework for Users' Privacy in Social Recommender Service

Ahmed M. Elmisery, S. Rho, Dmitri Botvich

In the ICTPS International Journal of Platform Technology (JPT), Volume 2, Issue 2, March 2014.

JOURNAL OF PLATFORM TECHNOLOGY VOL. 2, NO. 1, MARCH 2014

Holistic Collaborative Privacy Framework for Users' Privacy in Social Recommender Service

[1*]Ahmed M. Elmisery, [2,] S Rho, and [1]Dmitri Botvich
[1]*TSSG, Waterford Institute of Technology-WIT-Co. Waterford, Ireland*
ahmedmohmed2001@gmail.com
[2,] *Department of Computer Science, Kyungpook National University, Taegu, South Korea*

** Corresponding Author*

Abstract

Nowadays, it is crucial to preserve the privacy of end-users while utilizing a third-party recommender service within content distribution networks so as to maintain their satisfaction and trust in the offered services. The current business model for those recommender services is centered around the availability of users' personal data at their side whereas consumers have to trust that the recommender service providers will not use their data in a malicious way. With the increasing number of cases for privacy breaches of personal information, different countries and corporations have issued privacy laws and regulations to define the best practices for the protection of personal information. The data protection directive 95/46/EC and the privacy principles established by the Organization for Economic Cooperation and Development (OECD) are examples of such regulation frameworks. In this paper, we assert that utilizing third-party recommender services to generate accurate referrals are feasible, while preserving the privacy of the users' sensitive information which will be residing on a clear form only on his/her own device. As a result, each user who benefits from the third-party recommender service will have absolute control over what to release from his/her own preferences. To support this claim, we proposed a collaborative privacy middleware that executes a two stage concealment process within a distributed data collection protocol in order to attain this claim. Additionally, the proposed solution complies with one of the common privacy regulation frameworks for fair information practice in a natural and functional way - which is OECD privacy principles. The approach presented in this paper is easily integrated into the current business model as it is implemented using a middleware that runs at the end-users side and utilizes the social nature of content distribution services to implement a topological data collection protocol. We depicted how our middleware can be integrated into a scenario related to preserving the privacy of the users' data which is utilized by a third party recommendation service in order to generate accurate referrals for users of mobile jukebox services while maintaining their sensitive information at their own side. Our collaborative privacy framework induces a straightforward solution with accurate results which are beneficial to both users and service providers.

Keywords*: Privacy, Recommender service, Collaborative privacy*

1. Introduction

Different online social services have been developed since the last decade and they have had a profound effect on today's society. With the emergence of Web 2.0 and the spread of social media, there has been a growing demand of providing services that support social network platforms. Content distribution services are perpetually being deployed, where an increasing volume of personal data is being processed in return for personally tailored audio tracks, videos, and news. This personalization task is performed by a recommender system which might be running as a part of the content distribution service or as a third party service. In the first case, content distribution service providers are required to buy, build, train, and maintain their recommender system infrastructures despite exponential costs. Moreover, in order to run this service well, providers need to recruit a highly specialized team to tune and handle ongoing problems that arise once the service runs. However, in the second case, content distribution service providers could opt for the outsource service

model as it enables them to overcome their lack of computational power or expertise. They can plug in and subscribe to a third party service provider running the recommender service built on shared infrastructure via the Internet, where user's data is outsourced to this recommender service to perform the desired processing thereon. The recognition of the outscore service model is steadily increasing because it simplifies deployment and reduces client acquisition costs. Multitenancy feature of those online services permits content providers to scale as quick and as much as needed without replacing costly infrastructure or adding IT staff. Privacy is the main concern for theses online social service providers as service providers might be situated abroad with totally different legal structures and data privacy laws. In practice, users have shown an increasing concern for sharing their private data, especially in the case of untrusted parties [1]. As a result, the need to protect users' personal sensitive data is more crucial than ever as the users of these services have shown an increasing concern for exposing their personal data to untrusted entities so as to receive value-added services [1]. They need to realize full control over their sensitive data collected by these recommender services and cannot accept a compromise that their data might be fully accessible to an external party. This in most cases can forestall these users from fully embracing these content distribution services.

Privacy violations are prohibited in many countries. However, there is an absence of effective methods to enforce the law. This downside is exacerbated once information is used about individuals without their knowledge. As it should, if the customer has the proof that his/her privacy has been violated by the merchant, he could complain to the proper authorities, so that justice might be served. However, no amount of "justice" can fully restore his/her privacy. Two common means can be utilized for guaranteeing the privacy, technological, and legislation solutions. The former approach refers to technical methods and tools that are integrated into systems or networks to reduce the collection of accurate personal data. Such methods and tools are referred to as privacy enhancing technologies (PETs). One example of such PETs, which will be mentioned during this paper, is a middleware that executes topological formation for data collection along with a two stage concealment process that aims to control the amount of information the users reveal in the initial contact, and eliminates the necessity to release personal data in the raw form and permits the users to act anonymously. As for privacy legislation, it refers to data protection legislation restricting the gathering and usage of private personal data by data processors in order to define the best practices for the protection of personal information. Four examples for such privacy guidelines are the EU Directives 95/46/EC [2] and 2002/58/EC [3], UK's Data Protection Act and OECD privacy principles [4]. Despite the fact that several nations have developed privacy protection laws and regulations to guard against the secret use of personal information, the present laws and their conceptual foundations have become outdated because of the continuous changes in technology [5]. As a result, these personal data reside on databases of service providers, largely beyond the control of existing privacy laws, leading to potential privacy invasion on a scale never before possible. It is commonly believed that privacy is most successfully protected by a holistic solution that combines both technological and legislative efforts.

Among several existent approaches to recommender services that pride themselves in providing accurate recommendations, only a few tackle the privacy issues and aim to manage the privacy risk of social recommender systems as addressed by [6]. Most of the "privacy-concerned" social recommender services developed nowadays are either based on a trusted third-party model or on some generalized architecture. In order to use the service, the end-users have to divulge their personal data to the social recommender service and expect that the service providers will not use it in a malicious manner. Moreover, other systems address this problem with techniques to protect the processing of data stored on untrusted providers' systems. Besides, several of the existing recommender services which are based on multi-party recommendation protocols did not take into consideration the privacy issue. Therefore, our main challenge in this paper is to design an efficient privacy enhancing technology that shields against unauthorized access to the user's personal data, while at the same time exposing a sufficient amount of information to the third party recommender service in order to extract useful recommendations.

This paper presents a novel approach where sensitive data has two copies a concealed version, which is located on the recommender service side and a plain version that is stored

on the client side. Our approach for enhancing the users' privacy is to deploy a middleware on the client side where his/her data can be either kept private, or released in a locally concealed form. The latter implies that data is shared in a private manner after concealing it on the user's side using local concealment techniques/algorithms. We built a middleware that takes into consideration the social side during collecting users' data for these external recommender services. This middleware can be utilized for third party recommender services to facilitate access to a wealth of users' data in a privacy preserving manner. Our aim is not only limited to preventing the disclosure of sensitive data but also preserving the usefulness of data as much as possible to be only effective for the required computation. The rest of this paper is organized as follows. In Section 2, related works are described. Section 3 introduces OECD privacy principles and their implication in designing PET solutions. The proposed solution based on our collaborative privacy framework entitled *EMCP* (Enhanced Middleware for Collaborative Privacy) is introduced in Section 4. In Section 5, motivations and restrictions of the various prospective parties in our collaborative privacy approach are depicted in detail. Possible scenarios for the collaborative privacy framework were demonstrated in Section 6. In Section 7, the framework prototype is presented. Finally, the conclusion and future research are given in Section 8.

2. Related Work

There are many solutions in the literature that were proposed to achieve privacy in recommender systems. The work in [7] was the first proposal to attain this; it considers a scenario in which a centralized recommender system generates recommendations using the collaborative filtering approach. Users remove some selected parts from their profiles before sending them to the recommender. The recommender is able to attain recommendations because it was able to predict to some extent the missing parts. Attackers cannot learn the original ratings from the protected ones, but users can decide if their original ratings are included in the model using zero knowledge protocols. In this way, there is no external entity that has access to the private profile of a user. In [8] a privacy preserving approach is proposed based on peer to peer techniques using users' communities, where the community will have an aggregate user profile representing the group as a whole but not individual users. Personal information will be encrypted and the communication will be between individual users but not servers. Thus, the recommendations will be generated at the client side. In [9] a theoretical framework is proposed to preserve the privacy of customers and the commercial interests of merchants. Their system is a hybrid recommender system that uses secure two party protocols and public key infrastructure to achieve the desired goals. In [10, 11] another method is suggested for privacy preserving on centralized recommender systems by adding uncertainty to the data using a randomized perturbation technique while attempting to make sure that necessary statistical aggregates such as the mean don't get disturbed much. Hence, the server has no knowledge about the true values of the individual rating profiles for each user. They demonstrate that this method does not essentially decrease the obtained accuracy of the results. But recent research work in [12, 13] pointed out that these techniques don't provide the levels of privacy as was previously thought. In [13] it is pointed out that arbitrary randomization is not safe because it is easy to breach the privacy protection it offers. They proposed random matrix based spectral filtering techniques to recover the original data from perturbed data. Their experiments revealed that in many cases random perturbation techniques preserve very little privacy. Similar limitations were detailed in [12]. Storing the user's rating profiles on their own side and running the recommender system in a distributed manner without relying on any server is another approach proposed in [14], where authors proposed transmitting only similarity measures over the network and to keep users rating profiles secret on their side to preserve privacy. Although this method eliminates the main source of threat against the user's privacy, it requires higher cooperation among users to generate useful recommendations.

3. The OECD Privacy Principles

The organization for economic co-operation and development (OECD) [4] formulated sets of principles for fair information practice that can be considered as the primary components for the protection of privacy and personal data. A number of countries have adopted these principles as statutory law, in whole or in part in order to govern the data that customers outsource for third party services operating at remote sites. These principles can be described as follows:

- **Collection limitation**: Data collection and usage for a remote service should be limited only to the data that is required to offer an appropriate service.
- **Data quality:** Data should be used only for the relevant purposes for which it is collected.
- **Purpose specification:** Remote services should specify upfront how they are going to use the data and users should be notified upfront when a system will use it for any other purpose.
- **Use limitation:** Data should not be used for purposes other than those disclosed under the purpose specification principle without user consent.
- **Security safeguards:** Data should be protected with reasonable security safeguards (encryption, secure transmission channels, etc.).
- **Openness:** The user should be notified upfront when the data collection and usage practices started.
- **Individual participation:** Users should have the right to insert, update, and erase data in their profiles stored on remote services.
- **Accountability:** Remote services are responsible for complying with the principles mentioned above.

3.1. The Implications of OECD Principles in Designing an Efficient PET

In this section, we will investigate the research work in [15] that classifies the implications of the OECD principles with respect to designing an efficient PET. Then we will use their suggestions in order to state which of these principles should be considered as a norm in designing our proposed PET:

- **Collection Limitation**: This principle is ambiguous and it is difficult to be applied in our PET. The boundaries and content of what is considered private differ among cultures and individuals, but share basic common themes. Inspired from the work in [16], we summarized the challenges for this principle as boundaries and for each boundary, we describe a tension which the boundary has to face. These boundaries are as follows:
  - The Disclosure boundary (privacy and publicity) - we can define this as a tension between data elements that is private and public. The user has to decide what to keep private and what to make public.
  - The Identity boundary (self and other) - the users need to decide which identity to disclose to whom. So, here is a tension between different identities a user might have.
  - Temporal boundaries (Past, Present, and Future) - here is a tension on the time aspect. What is not private in the past might become in the future and vice versa and also when the information is being persistent much of the actions done in the past cannot be undone.

  Our contributions in this research address the first two boundaries. As a result, the end-users have the choice to determine a sensible realization for the notion of very sensitive data. Moreover, they are responsible for making their data public or private by employing privacy preferences languages to specify rules or levels for releasing their data such that a conscious automatic choice can be made about which group gets to see what. Also, catering to the second boundary, giving the end-users the choice to join a peer-group, using an anonymous network or leaving the recommendation process, where the users can join a peer-group only with trusted end-users or their friends. However, the temporal boundary is not really addressed in this paper, but we plan to address it in future research.
- **Data Quality Principle**: Most of the proposed PETs assume that the data is in an appropriate form to be processed by the current obfuscation and/or anonymization techniques. However, data cleaning methods could be utilized locally to handle imprecision and errors in data before

any concealment process. We mitigated this principle by selecting two common types of erroneousness in the users' data, which are incomplete users' profiles and outliers. Then after, we proposed a set of concealment algorithms which take into consideration pre-processing the incomplete user profiles and handling outliers on these profiles. Other types of deviations should be investigated in future research. Meanwhile, we left the task of handling other erroneousness to the user, in order to maintain an accurate profile for the recommendation process and to facilitate a straightforward concealment process.

- **Purpose Specification Principle:** This principle is relevant for our PET; users should be well informed at the outset prior to the collection and processing of their information.
- **Use Limitation Principle**: This principle is relevant for our PET and related to the previous principle. The gathered information from users must be used only for the purpose that was disclosed at the time of collecting it.
- **Security Safeguards Principle**: This principle is relevant for our PET but related in general to data security. We have mitigated this principle by proposing a middleware that runs at the user side and assures anonymity and privacy of each individual user. Within this approach, the proposed middleware assigns two profiles per each user, one is a local profile in a plain form and it is stored locally on the user machine and the other is a public profile that represents the local profile in a concealed form and it is ready to be released for recommendation purposes. This approach ensures that the users' personal data are protected from malicious attackers.
- **Openness Principle**: This principle is relevant for our PET; users should know what data about them has been gathered and being processed. However, most of the social recommender services do not disclose the logic behind the scene due to intellectual property issues. We have mitigated this principle by enabling the user to decide either to join or not a certain recommendation process and also to control what data to be released for a certain recommendation process.
- **Individual Participation Principle**: This principle is relevant for our PET; users are aware that the generated referrals are related to their released data. Users can challenge the value of the offered referrals and decide either to participate or not. Therefore, there should be a certain mechanism to carefully outline the weight of this principle to the users.
- **Accountability Principle**: This principle is irrelevant for our PET; remote services should inform users about the policies related to the usage of the generated recommendation model including the consequences of abusing the collected data. This principle is too general in scope or area to be utilized for PETs.

Based on the outline we declared above, we categorized the OECD principles into two groups according to their influence on the context of designing our proposed PET:

- **Group 1**: Consists of those principles that should be considered as design principles in our proposed PET, such as data quality, purpose specification, use limitation, security safeguard, openness, and individual participation.
- **Group 2**: Involves some principles that are too general or irrelevant in PETs. Some of those principles depend on the applications where PETs are needed, and their effects should be understood and carefully evaluated depending on these applications.

The principles categorized in groups 1 are relevant in the context of our collaborative privacy approach and are fundamental for further research, development, and deployment of PETs.

4. Collaborative Privacy Framework using EMCP for Third-Party Social Recommender Service

EMCP has been proposed to satisfy the privacy requirements of privacy aware users. In our earlier work presented in [17-20], the proposed collaborative privacy framework has implemented a two stage concealment process, where each stage utilizes a set of machine learning based stochastic techniques that introduce carefully -chosen artificial noise in the data so as to retain its statistical content while concealing all private information, in that way privacy is achieved for both individual participants and groups of participants. The following terms will be used during the remaining parts of this paper:

1. User's profile refers to the personal information and preferences for individual system users. The personal information corresponds to any personally identifiable information such as name, gender, zip code, age, address, etc., while preferences correspond to the consumed items with their ratings where these ratings are referring to which degree an item was interesting to this user.
2. An individual user is a registered customer/client for the content distribution service. We referred to a user who is requesting recommendation as the target user while users who are willing to participate in a recommendation process are referred to as participants.
3. The third party entity that offers the recommendations/referrals was referred to as the social recommender service while the entity that delivers the aforementioned recommended contents was referred to as the content distribution service.
4. Both the users and the content distribution service can be called clients for the social recommender service, where each social recommender service can serve multiple content distribution services with their users using a service-oriented infrastructure.

Each individual user who utilizes the recommendation of the content distribution service is hosting and running the *EMCP* middleware within his/her personal device. *EMCP* is the main architectural element of our collaborative privacy framework, such that *EMCP* is responsible for executing the topological formation protocol for data collection and providing controlled access over what personal information is to be released with a different degree of granularities to external parties. The content distribution service uses *EMCP* to manage and store the users' profiles while using the content delivery network of their service. The main characteristics of our *EMCP* middleware architecture is:

- Form the content distribution service's point of view, *EMCP* is a decentralized system for the storage and management of users' profiles.
- Form the user's point of view, *EMCP* is a centralized system where all his/her personal information and preferences are stored locally on his/her personal device.

As we mentioned earlier, the proposed collaborative privacy framework was implemented using *EMCP* middleware which combines all of these techniques to make it possible to efficiently take advantage of this work. *EMCP* enables participants to be organized on a distributed topology during data collection, where participants are organized into peer-groups and each peer-group contains a reliable peer to act as a trusted aggregator that is an entitled super-peer who will be responsible for anonymously sending the aggregated data of members within this peer-group to the social recommender service. Additionally after receiving the referrals list, the super-peer will be responsible for distributing this list back to its peer-group. Electing these super-peers is based on negotiation between participants and a trusted third party; this trusted third party is responsible for generating certificates for all participants, and managing these certificates. In addition, it is responsible for making assessments on those super-peers according to participants' reports and periodically updates the reputation of those super-peers.

Utilizing topological formation within our collaborative privacy framework attains privacy for participants with relatively low accuracy lose. Moreover, it prevents the service provider from creating a centralized database with a raw personal data for each user. Additionally, it permits a decentralized execution of a two-stage concealment process on the users' personal data that satisfies the requirements of high scalability and reduces the risk of privacy breaches. The formation of these peer-groups is done through a specific virtual topology in order to create an aggregated profile (group profile). This topology might be simple like a ring topology or complex like hierarchical topology (see Figure 2). This ordering enables users to attain privacy by collaboration between them. Data is shared between various users within the same peer-group after it is locally concealed based on the trust level. The super-peer will be responsible for executing a global concealment process on the aggregated profile (group profile) before delivering it to the recommender service. In this approach, the notion of privacy surrounding the disclosure of the users' preferences and the protection of trust computation between different users are together the backbone of this framework. Trust based concealment mechanism was applied at the participant side such that trust computation is done locally over the concealed participant's preferences. Utilizing

trust heuristic as input for both group formation and the local concealment process has been of great importance in mitigating some of the malicious insider attacks described in [21] and maintains an optimized utility for the concealed data [18].

The two stage concealment process with *EMCP* executes a set of newly proposed stochastic techniques for concealing users' personal data which are released to recommendation requests. This is not a straightforward task as the two stage concealment process should make sure that the concealed data is still useful for the recommendation phase, which usually requires that changes on the users' personal data be as small as possible. However, users' profiles are complicated and are an interrelated structure. Making small changes on it could cause an unexpected influence on the overall recommendation process. The proposed techniques combine approaches from the machine learning clustering analysis that consider knowledge representation in the domain of data privacy in order to preserve the aggregates in the dataset to maximize the usability of this data, with a view to accurately perform the desired recommendation process. The validity of the framework is demonstrated by the implementation and evaluation of the proposed solution within a set of important innovative applications. A general overview on the proposed framework is shown in Figure 1.

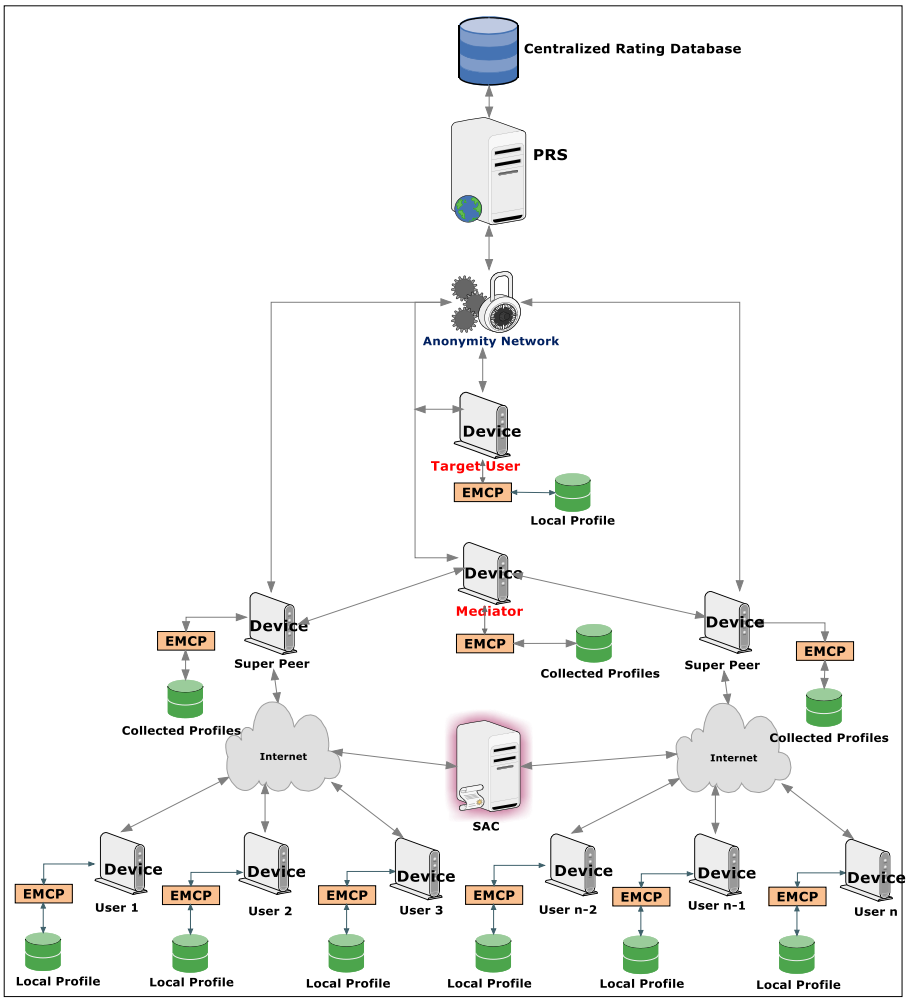


Figure 1. EMCP Middleware in Third-Party Social Recommender Service.

As a result, the proposed collaborative privacy framework attains anonymity and privacy. The anonymity is achieved by utilizing pseudonyms by either running the communication through an anonymity network like Tor or by a topological formation that divides users into a coalition of peer-groups, whereas each peer-group is to be treated as one entity by aggregating its members' data in one aggregated profile at the super-peer and then this

super-peer will handle the interaction with the social recommender service. Individual participants might benefit from this anonymity while interacting with the recommender. If profiles cannot be identified and assuming that the initial user cannot be traced back, the system protects the privacy of the users even if the profiles are sent in clear. However, participants' data privacy is achieved as each participant within the peer-group performs at least one stage in the concealment process based on his/her role in the peer-group. Traditional members perform a local concealment process before releasing their data to external entities. Local concealment is a pre-processing step that is based on clustering the sensitive data then applies a concealment algorithm on the extracted partitions, so as to take into consideration the correlations and range of different data cells within sensitive data. The super-peers of every peer-group aggregates the data received from traditional members to form a group profile then execute a global concealment process on the group profile before releasing it to the service provider. This sort of two stage concealment process enforces anonymity for participants' identities and privacy for their data.

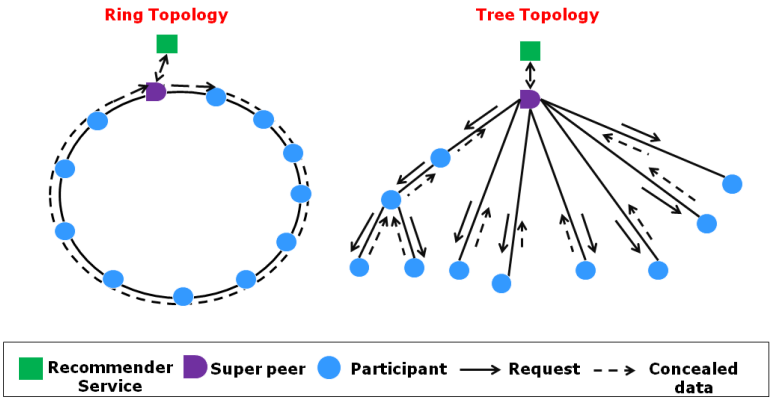


Figure 2. Topology for creating aggregated profile in Peer-groups

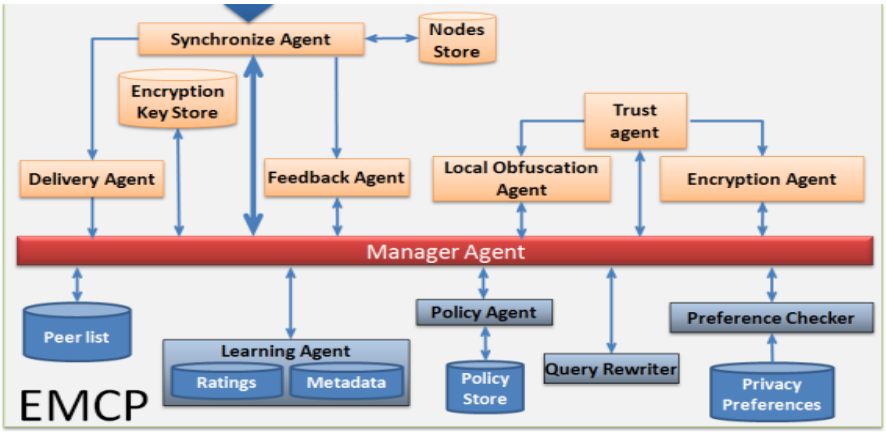


Figure 3. Inside *EMCP* Components

4.1. Design of *EMCP* Middleware

Figure 3 illustrates the components of the proposed enhanced middleware for collaborative privacy (*EMCP)* running inside the user's local device, which in an earlier version was called (AMPR). *EMCP* consists of different co-operative agents. A learning agent captures user interests about miscellaneous items explicitly or implicitly to build a rating database and meta-data database. The local obfuscation agent implements a local concealment process to achieve user privacy while sharing his/her preferences with super-peers or the external social recommender service (PRS). The encryption agent is only invoked if the user is acting as a super-peer in the recommendation process; it executes global concealment on the aggregated profile (collected profiles from the members of the

peer group). The two stage concealment process acts as wrappers to conceal preferences before they are shared with any external social recommender service.

Since the database is dynamic in nature, the local obfuscation agent periodically conceals the updated preferences, and then a synchronize agent forwards them to the social recommender service (PRS) upon owner permission. Thus, recommendation can be made on the most recent preferences. Moreover, the synchronize agent is responsible for calculating and storing parameterized paths in an anonymous network that attain high throughput, which in turn can be used in submitting preferences anonymously. The policy agent is an entity in *EMCP* that has the ability to encode privacy preferences and privacy policies as XML statements depending on the host role in the recommendation process. Hence, if the host role is as a "super-peer", the policy agent will has the responsibility to encode data collection and data usage practices as P3P policies via XML statements which are answering questions concerning the purpose of collection, the recipients of these profiles, and the retention policy. On the other hand, if the host role is as a "participant", the policy agent acquires the user's privacy preferences and expresses them using APPEL as a set of preferences rules which are then decoded into a set of elements that are stored in a database called "privacy preferences" in the form of tables called "privacy meta-data". These rules contain both a privacy policy and an action to be taken for such a privacy policy, in such a way this will enable the preference checker to make self-acting decisions on objects that are encountered during the data collection process regarding different P3P policies (e.g.- privacy preferences could include: certain categories of items should be excluded from data before submission, expiration of purchase history, usage of items that have been purchased with the business credit card and not with the private one, generalize certain terms or names in the user's preferences according to defined taxonomy, using synonyms for certain terms or names in the user's preferences, suppressing certain items from the extracted preferences, and insert dummy items that have the same feature vector like the suppressed ones as described in [22], limiting the potential output patterns from extracted preferences etc. in order to prevent the disclosure of sensitive preferences in the user's profile). Query Rewriter rewrites the received request constrained by the privacy preference for its host.

4.2. The Interaction Sequence between Parties within Collaborative Privacy Framework

Figure 4 shows the participants interactions with super-peers and third-party social recommender service. A general overview of the recommendation process in the proposed framework operates as follows:

1. The target user (user requesting recommendations) broadcasts a message to other users in the network requesting a recommendation for a specific genre or category of items. Thereafter, the target user selects a set of his/her preferences to be used later in the computation of the trust level at the participant side. The local obfuscation agent is employed to perform the local concealment process on the released data. Finally, the target user dispatches this data to the individual users who have decided to participate in the recommendation process.
2. Each group member negotiates with the security authority centre (SAC) to select a peer with the highest reputation to act as a "super-peer" which will act as a communication gateway between the recommender service and the participants in its underlying peer-group. SAC is a trusted third party responsible for making an assessment on those super-peers according to the member' reports and super-peer-reputations.
3. Each super-peer negotiates with both the target user and the recommender service to express its privacy policies for the data collection and usage process via P3P policies.
4. At the participant side, the manager agent receives the request from the target user along with the P3P policy from the elected super-peer; then it forwards this P3P policy to the preference checker and the request to query rewriter. The preference checker ensures that the extracted preferences do not violate the privacy of its host which were previously decaled by the use of APPEL preferences. The query rewriter rewrites the received request based on the feedback of the preference checker. The modified request is directed to the learning agent to start the collection of preferences that could satisfy the modified

query and forwards it to the local obfuscation agent. Finally, the policy agent audits the original and modified requests plus estimated trust level and P3P policy with previous requests in order to prevent multiple requests that might extract sensitive preferences.

5. The trust agent calculates approximated interpersonal trust between its host and the target user based on the received preference. It is done in a decentralized fashion using the entropy definition proposed in [23] at each participant side. The trust agent sends the calculated trust value to its pre-specified super-peer. The estimated trust values are forwarded to both the super-peers and the social recommender service. Then after, the locally concealed data for each participant is sent to the super-peers of their pre-specified peer-group.
6. Upon receiving the locally concealed preferences from each participant, each super-peer filters the received preferences based on the trust level. Then, each super-peer builds a group profile (aggregated profile) in order to perform the global concealment process on this profile. The super-peer can seamlessly interact with the social recommender service (PRS) by posing as a user and has a group profile as his/her own profile.
7. The social recommender service (PRS) runs the recommendation algorithm on the received aggregated profile then forwards the generated referrals list along with the predicated ratings to each super-peer in the peer-group. Super-peers publish the final list to the target user and/or participants. Finally, each participant report scores about the elected super-peer of his/her peer-group and the target-user to SAC, which helps to determine the reputation of each entity involved in the referrals generation.

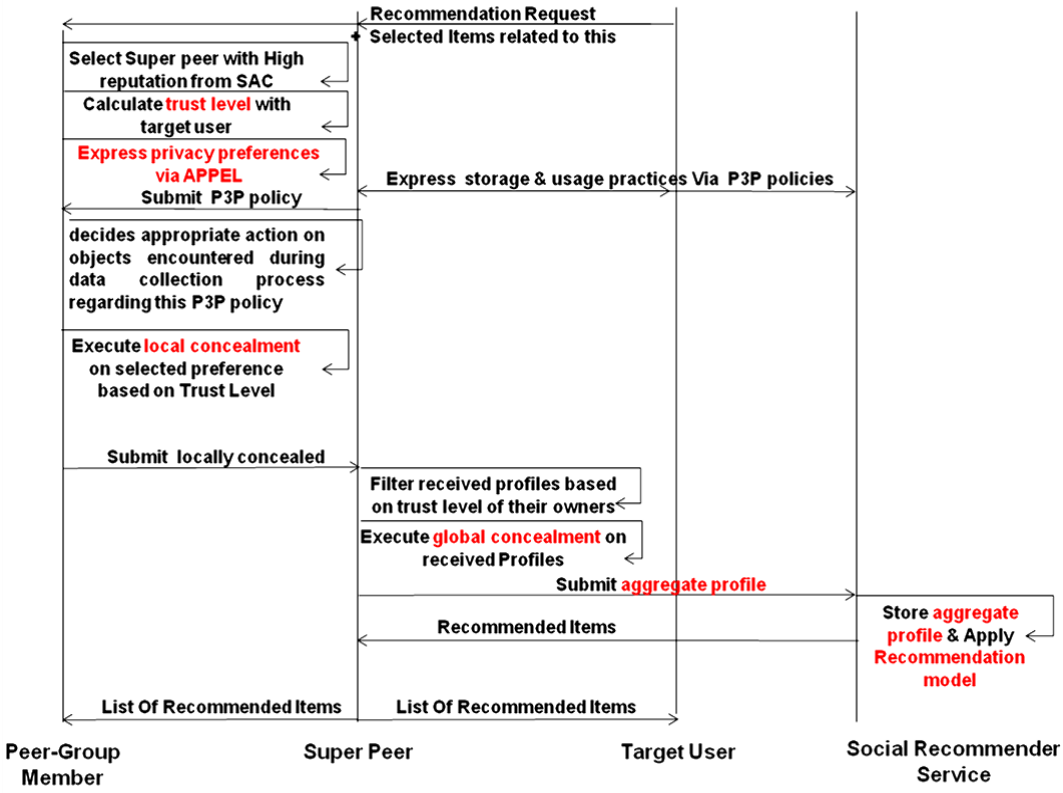


Figure 4. Interaction Sequence Diagram for Collaborative privacy framework

In order to demonstrate the applicability of this framework, this research presented a case study focusing on mobile jukebox service. This scenario is motivated by protecting the privacy of users' profiles while utilizing the jukebox service and its implications. A typical user profile with this service contains the user's personal information along with his/her musical tastes and listening habits. The reason for selecting this case study was due to the fact that it represents the more pressing issue on privacy research and we hoped to enable the deployment of privacy-aware mobile jukebox recommender service using the

collaborative privacy approach. Obviously, other practical scenarios still exist for the proposed framework. However, in this research we are unable to address all of them.

4.3. The Role of OECD Principles in the Collaborative Privacy Framework

OECD principles rely on the commitment of service providers on revealing their data handling practices accurately. However, the current perspective illustrates that it is likely for them to not follow these principles in full. We have utilized the OECD principles as design guidelines for our collaborative framework. The role of OECD principles in designing our proposed PET will be outlined in this subsection, where we have termed the proposed PET in this research as an enhanced middleware for the collaborative privacy framework which is abbreviated as *EMCP*. The proposed framework reduces privacy risks and facilitates privacy commitment. Moreover, it realizes privacy aware recommendations while complying with the current business model of third-party social recommender service. The privacy obtained through the proposed collaborative privacy approach is as follows:

- **Collection Method**: The proposed solution attains an explicit data collection mode. Users are aware that a data collection within a recommendation process is happening and they can make a wise decision about whether or not to provide their data in this recommendation process. Privacy policies such as P3P are utilized to explain to the users how their data is going to be used. Users utilize privacy preferences in order to control what data from their profiles gets collected in which concealment level. However, formalizing such privacy preferences is not an easy task. Users need to realize various privacy issues. Additionally, users need to deduce future recommendation requests that might raise privacy concerns for his/her collected data. The user can employ an anonymous network while sending this locally concealed data to either the super-peer or the social recommender service.
- **Duration:** The proposed solution attains a session based collection that allows for a simpler service that does not need the storage and retrieval of users' profiles. The data related to the recommendation process is collected from users' profiles in a concealed form. This concealed data is only feasible for recommendation purposes. This reduces privacy concerns as minimal data to be collected and also ensures the compliance with privacy laws. The concealed data is stored at the third party service in order to enhance the recommendation model and future requests. Moreover, this data by default is protected by the privacy protection laws.
- **Initiation:** The proposed solution attains a user based recommendation. Users are the entities that initiate the recommendation process; each user in the network is aware that a recommendation process is happening and he/she can decide whether or not to join it. The incentive for participants when joining a recommendation request includes receiving referrals regarding a certain topic in a private manner.
- **Anonymity:** The proposed solution attains anonymity which aids in preventing frauds and sybil attacks. The anonymity is realized within the collaborative privacy framework using the following procedures:
  - **a.** Dividing system users into a coalition of peer-groups: each peer-group to be treated as one entity by aggregating its members' concealed data in one aggregated profile at the super-peer, then this super-peer will handle the interaction with the social recommender service. Participants within the coalition interact with each other in a P2P fashion and form a virtual topology to aggregate their data.
  - **b.** Using anonymous channels like Tor: Individual participants might benefit from these anonymous channels while contacting the recommender service or other members in their coalition.
  - **c.** Utilizing pseudonym for users: each user within the system is identified by a pseudonym in order to reduce the probability of linking his/her collected profiles' data with a real identity.
- **Local Profiles:** Our solution attains local profiles storage. Users' profiles are stored locally on their own devices (Setup box, Smart phone, Laptop...) in an encrypted form. This can guarantee that these profiles are attainable only to their owners. Furthermore, in doing so these profiles will be inaccessible to viruses or malware that may affect the user's machine to gather his/her personal data. As a result, each user will possess two profiles; one is a local profile in a

plain form that is stored locally in his/her machine and it is updated frequently. The other is a public profile in a concealed form that is stored remotely at the service provider and it is updated periodically within each recommendation process where this user participated.

- **Stochastic Techniques for Data Privacy:** Our solution relies on a set of machine learning cluster analysis based stochastic techniques. These techniques are to be carried out in two consecutive steps within a two stage concealment process. The proposed techniques destroy the structure in data but, at the same time, maintain some properties in it which is required in the planned recommendation. Additionally, the implementation of such applications confirmed that is feasible to make use of and, at the same time, to protect the personal sensitive data of individuals, and do so in an accurate way.

4.4. Privacy Management Approach using the Collaborative Privacy Framework

The core hypothesis of our collaborative privacy approach is that personal profiles are stored locally at the users' side of their personal device. Two related questions may arise in the mind; how can we ensure that the end-users will participate in such a solution and what are the incentives for service providers to adopt this solution. We are aware that our collaborative privacy approach represents an extreme case for privacy management and enforcement. However, our collaborative privacy approach serves as a proof of the concept that fair information practices can be deployed, implemented, and enforced in a more efficient way when it is being utilized in service oriented architecture like mobile jukebox service rather than adopting the current approaches. In particular, within our framework, personal users' profiles can be handled in a privacy respecting manner that is complying with the OECD privacy principles. The recent emergence and spread of user centric applications, makes it feasible to fully embrace a privacy enhanced technology (PET) such as our collaborative privacy framework. Nevertheless, the growing privacy invasions within the current approaches have contributed in facilitating the misuse of personal information, which is considered one of the most common problems when taking advantage of digital services.

Due to the previously mentioned reasons, we believe there are some shortfalls in separating technological and legislative solutions, which open the doors for us to further investigate into a new holistic solution that combines both technological and legislative efforts together in a unified framework. The new solution meets the crucial requirements of OECD privacy principles and amends the user's control over his/her personal information that is released to external parties. In this regard, we developed and evaluated our collaborative privacy framework in different scenarios. Obviously, that much work has to be done in order to demonstrate the possibility of applying a solution like *EMCP* in the various business models while complying with varied privacy guidelines. However, our previous research work confirms that our collaborative privacy framework is feasible for different applied contexts.

5. Motivations and Restrictions of the Various Prospective Parties in our Collaborative Privacy Approach

There are numerous motivations and restrictions for the various parties involved within our collaborative privacy framework, which make it not only valuable to the user but also to service providers. Our proposed middleware which is employed in the implementation of the framework permits the end-users to control the privacy of their released data while interacting with third-party social recommender services. This kind of approach is quite flexible and can easily be adopted in a conventional business model of the current service oriented based services, like social recommender services because it is executed at the user side and it takes advantage of the social structure that is offered by the online content distribution service without the need for significant modifications at the service provider side. Moreover, service providers can also attain many benefits from adopting the proposed framework, such as, promoting a privacy friendly environment for their offered services, simplifying the data management process at their side and finally reducing their liability to secure their clients' personal information.

5.1. Motivations and Restrictions for Users

Users' Motivations

- Attaining ultimate control over their personal information: The users can determine for each recommendation request, what super-peers and purposes their data will be released for, and what data from their profiles gets collected in which concealment level. Additionally, they are aware of how long this data will be retained at external parties.
- Utilizing up-to-date data for recommendations purposes: Storing the data locally at the user side facilitates the creation of accurate profiles and simplifies the update of these profiles with the most recent consumption history of these users. As a result, each time a recommendation request occurs, the users will release updated data from their current profile instead of using outdated data stored at the social recommender service, which will allow generating accurate referrals that match their changing preferences and tastes.
- Specifying their privacy preferences: Users can express their privacy preferences using APPEL as a set of rules which are then decoded into a set of elements that are stored in a privacy preferences database. These rules will enable *EMCP* to make self-acting decisions on data elements that are encountered during the data collection process regarding different P3P policies.
- Reducing the impact of privacy breaches: In case the occurrence of privacy invasion happens at the social recommender service, the leaked users' data will be worthless with a diminished informative value, because it is already concealed with a two stage concealment process and cannot be linked directly to a specific user within a peer-group. Moreover, the leaked users' data is concealed in a way to be only useful for recommendations purposes and it would be difficult to perform different kinds of analytical processes on such data.
- A third option for privacy aware users: Privacy aware users will no longer have to choose between two options, either releasing their whole data to a recommender service which they have to trust or not using the service at all. Our collaborative privacy framework provides an alternative to the current models of practice.

Users' Restrictions

- The users have to formalize their privacy preference, which is a critical task, as the users need to realize various privacy concerns. Additionally, they need to deduce future recommendation requests that might raise privacy concerns for their collected data.
- The collaborative privacy framework does not fully protect users from malicious super-peers. The malicious super-peer can uncover the user's anonymity during the release of his/her data to a specific recommendation request. This problem has been mitigated by utilizing anonymity networks while sending the data from users to super-peer and employing reputation mechanisms in order to select proper super-peers with a stable success rate. Moreover, the user's data is not in a raw form and its privacy is already protected with a local concealment process before leaving the user's device.

5.2. Motivations and Restrictions for Recommender Service Providers

Service Providers' Motivations

- Providing accurate referrals: The referrals are extracted from up-to-date data, which is collected prior to the start of the recommendation process. This has a number of beneficial advantages on the offered service, such as, reducing the users' frustration, increasing the number of potential users for the service, and raising the revenue of the service providers.
- Using the current social recommendation techniques: adopting the collaborative privacy framework does not require the design of new recommendation techniques, the current off-the-shelf social recommendation algorithms can be used directly on the concealed data without the need to return it back into a raw form.
- Readiness to be used in the conventional business model of the current service oriented based services: Most of the existing service providers find difficulties in integrating privacy enhancing technologies within their service, as the addition of privacy and cryptography components requires a significant change on their service's back- end

infrastructure. Our collaborative privacy framework utilizes the user and social sides of the service providers as an infrastructure for the implementation of our framework. The collaborative privacy framework is quite flexible and can easily be adopted in the current business model of social recommender services because it is executed at the user side and it takes advantage of the social structure that is offered by their service without the need for significant modifications at the service provider side.

- Simplifying the data management process at the service side: Within the collaborative privacy framework, the users' profiles are stored on their side on their own devices. However, in order to enable the service providers to use the users' data in more sophisticated business processes, a concealed public version of users' profiles are stored on their side to serve the enterprise business' initiatives of these service providers.
- Promoting a privacy friendly environment for the offered referrals: Privacy aware users will be encouraged to participate on such service, as their personal data will be stored locally on their own side and they can decide what data to be released for every request. Additionally, the released data will not leave their devices until it is properly concealed.
- Reducing the liability of service providers in securing their clients' personal information: The responsibility of the service providers for protecting their clients' personal data is alleviated, as the clear and accurate version of users' profiles are stored on the users' devices. Privacy invasion on these public profiles will not be as harmful as much as it is when compared with the ones that occur in the current conventional approaches of privacy.
- Enhance the efficiency of the content distribution providers: The extracted recommendations can be used to support the content distribution providers from different perspectives, such as maximizing the precision of target marketing and improve the overall performance of the current distribution network by building up an overlay to increase content availability, prioritization and distribution based on the predicated recommendations.

Service Providers' Restrictions

- Losing the control over users' profiles: Indeed, the users' profiles are stored remotely at their side, however, the service providers are also holding and storing public profiles from previous recommendation processes. Although, the public profiles are an outdated snapshot of the users' data in a concealed form, they are sufficient enough for training, building, and maintaining the recommendation model.
- Potential abuse for the service by malicious users: The anonymity attained by our collaborative privacy approach can induce malicious users to perform attacks on the service or other users while exploiting the advantage of hiding their identity, thus they can escape from legal prosecution. We have introduced the usage of security authority centre (SAC), which is a trusted third party responsible for assessing the reputation of each entity involved in the referrals generation process. Moreover, SAC is in charge of issuing anonymous credentials for each user in the system. Future research should investigate how to attain the functionality of SAC in P2P fashion and without relying on a centralized entity.

5.3. Privacy Enforcement

Utilizing topological formation for data collection with a two stage concealment process within our framework allows the user to control what data from their profiles gets collected and in which concealment level. Specifically, the public group profile that is exposed to the third party social recommender services contains a set of collected items from the users' profiles that are released to a specific recommendation request. These items usually represent a small proportion of items in relative relation to the total number of consumed items in the users' profiles. Moreover, the anonymity and concealment techniques used during the data collection process ensure attaining an appropriate privacy level for system users. Those are very important aspects in our framework that depicts its ability to diminish the impact of the privacy breaches, limit the misuse of personal information, and to enforce and verify the attained privacy for its users. Moreover, using P3P policies enable the user to

present evidence that his/her preferences were released for a specific recommendation process, at a specific time, and for a specific super-peer.

6. Prospective Scenarios for the Collaborative Privacy Framework

The proposed framework was utilized in diverse scenarios to create privacy aware versions for three beneficial applications of the social recommender service, which are a recommender service for IPTV content providers, data mash-up service for IPTV recommender services, and community discovery & recommendation service. Privacy aware versions of location based recommendation service and mobile jukebox content recommender service were also introduced in order to show the applicability of our approach. The implementation and evaluation of such applications of the collaborative privacy framework confirmed that it is possible to employ the personal profiles of users while preserving their privacy. In the next subsection, we will present a case study for mobile jukebox recommender service and how our collaborative privacy framework can be used as a privacy preserving infrastructure to control the privacy for users within the recommendation process.

6.1. Case Study: Mobile Jukebox Recommender Service

We consider the scenario where a social recommender service (PRS) is implemented on an external third party server and end-users give information about their preferences to that server in order to receive music recommendations. The user preferences are stored in his/her profile in the form of ratings or votes for different items, such that items are rated explicitly or implicitly on a scale from 1 to 5. An item with a rating of 1 indicates that the user dislikes it while a rating of 5 means that the user likes it. The recommender service collects and stores different users' preferences in order to generate useful recommendations.

In this scenario there are two possible ways for the user's discloser: through his/her personal preferences included in his/her profile [24] or through the user's network address (IP). *EMCP* employs two principles to eliminate these two disclosure channels, respectively. The two stage concealment process was used to conceal user's preferences for different items in his/her profile and an anonymous data collection protocol is used to hide the user's network identity by routing the communication with other participants through relaying nodes in Tor's anonymous network [25]. We didn't assume the server to be completely malicious. This is a realistic assumption because the service provider needs to accomplish some business goals and increase its revenues. In this scenario, we will use the mobile phone storage to store the user profile. However, the mobile jukebox recommender service maintains a centralized rating database for storing the group profiles that is used in model building. Additionally, we alleviate the user's identity problems stated above by using anonymous pseudonyms identities for users. The recommendation process based on the two stage concealment process in our framework can be summarized as follows:

1. The learning agent collects user's preferences about different items which represent a local profile. The local profile is stored in two databases, the first one is the rating database that contains (id, rating) and the other one is the metadata database that contains the feature vector for each item (id, feature1, feature2, feature3). The feature vector can include: genre, author, album, decade, vocalness, singer, instruments, number of reproductions, and so on.
2. The target user broadcasts a message to other users near him/her to request recommendations for a specific genre or category of items. Individual users who decide to respond to that request perform the local concealment process to conceal a part of their local profiles that match the query. The group members submit their locally concealed profiles to the requester using an anonymized network like TOR to hide their network identities.
3. After the target user receives all the participants' profiles (group profile), he/she executes a global concealment process to conceal the group profile. Then he/she can interact with the recommender service by acting as an end-user and have the group profile as his/her

own profile. The target user submits the group profile through an anonymized network to the mobile jukebox recommender service in order to attain recommendations.

4. The mobile jukebox recommender service performs its filtering techniques on the group profile which in turn return a list of items that are correlated with that profile. This list is encrypted with a private key provided by the target-user and it is sent back on the reverse path to the target user that in turn gets decrypted and published anonymously to the other users that participated in the recommendation process.

- ***Local Concealment Process using Clustering Transformation Algorithm (CTA).***

We have proposed a novel algorithm for the local concealment process in order to conceal the user's profile before sharing it with other users. CTA is designed especially for the sparse data problem we have here. CTA partitions the user profile into smaller clusters and then pre-processes each cluster such that the distances inside the same cluster will be maintained in its concealed version. We use local learning analysis (*LLA*) clustering method proposed in [26] to partition the dataset. After completion of the partitioning, we embed each cluster into a random dimension space (based on parameter *d-dim*) so the sensitive ratings will be protected. Then, the resulting clusters will be rotated randomly. In such a way, CTA conceals the data inside user's profile while preserving the distances between the data points to provide highly accurate results when performing recommendations. More details about the algorithm can be found in [27].

- ***Global Concealment Process using the Enhanced Value-Substitution (EVS) Algorithm.***

After executing the local concealment process, the global concealment phase starts. The key idea for EVS is based on the work in [28] that uses the Hilbert curve to maintain the association between different dimensions. In this subsection, we extend this idea as following: we also use the Hilbert curve to map *m-dimensional* profile to *1-dimensional* profile then EVS discovers the distribution of that *1-dimensional* profile. Finally, we perform perturbation based on that distribution in such a way to preserve the profile range. More details about the algorithm can be found in [27].

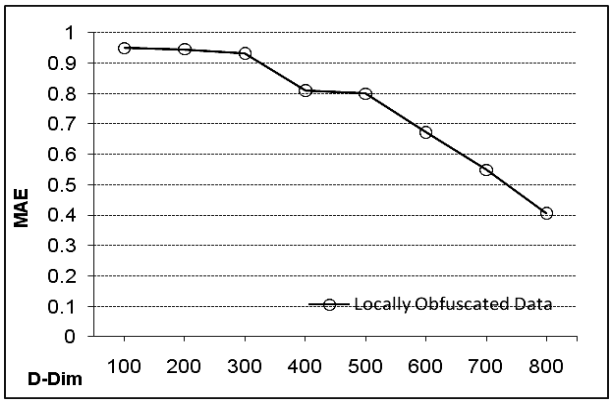


Figure 5. Accuracy of recommendations for the concealed dataset using CTA

- ***Experiential Results***

To evaluate the accuracy of CTA algorithm with respect to a different number of dimensions in the user profile, we controlled the *d-dim* parameters of CTA to vary the number of dimensions during the local concealment process. Figure 5 shows the performance of recommendations of locally concealed data in terms of mean absolute error (*MAE*), as shown in the accuracy of recommendations based on the concealed data is a little bit low when *d-dim* is low. But at a certain number of dimensions (500), the accuracy of recommendations on the concealed data is nearly equal to the accuracy obtained using the original data. In the second experiment performed on the CTA algorithm, we examined the effect of the *d-dim* on privacy level attained in terms of the variation of information (VI) metric. As shown in Figure 6, privacy levels decrease with respect to the increase in *d-dim* values in the user profile. The *d-dim* is the key element for controlling the privacy level where smaller *d-dim* value, the higher privacy level of CTA. However, clearly the highest privacy is at *d-dim*=100. There is a noticeable drop of attained privacy when we change *d-*

dim from 300 to 600. The *d-dim* value 400 is considered as a critical point for the privacy. Note that rotation transformation adds an extra privacy layer to the data and in the same time maintains the distance between data points to enable the recommender service to build an accurate recommendation model.

In this last experiment which was performed on the EVS algorithm, we measured the relation between different Hilbert curve parameters (order and step length) on the accuracy and privacy levels attained. We mapped the locally concealed dataset to Hilbert values using order 3, 6, and 9. We gradually increased the step length from 10 to 80. Figure 7 shows the accuracy of recommendations based on the different step length and curve order. We can see that as the order increases, the concealed data can offer better predictions for the ratings. This is because as the order has a higher value, the granularity of the Hilbert curve becomes finer. So, the mapped values can preserve the data distribution of the original dataset. However, selecting a larger step length increases the accuracy values as large partitions are formed with a higher range to generate random values from it, such that these random values substitute real values in the dataset.

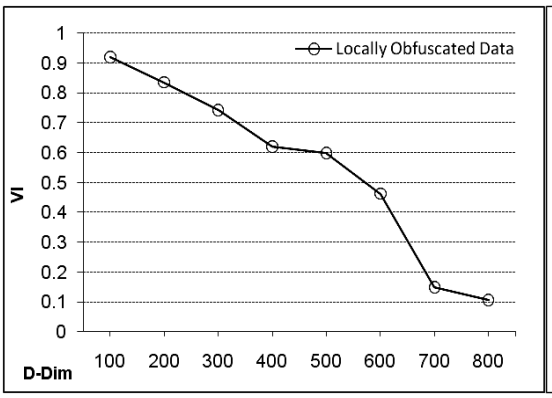


Figure 6. Privacy levels for the concealed dataset using CTA

Figure 7. Accuracy level for different step length and orders for EVS

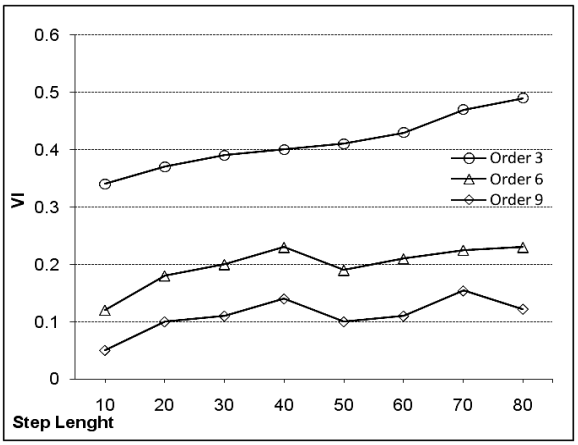


Figure 8. Privacy level for different step length and orders for EVS

Finally, as shown in Figure 8 when the order increases a smaller range is calculated within each partition which introduces less substituted values compared with lower orders that attain higher variation of information (VI) values. The reason for this is that the larger order divides the *m-dimensional* profile into more grids, which makes Hilbert curve better at reflecting the data distribution. Moreover, we can see that for the same Hilbert curve order the VI values are generally the same for the different step length except for order 3, in which VI values have a sharp increase when the step length grows from 50 to 60. The effect of increasing step length on VI values is more sensible in lower curve orders as fewer girds are formed and the increase of the step length covers more portions of them, which will introduce a higher range to generate the random values from it. So the target user should

select EVS parameters in such a way as to achieve a trade-off between privacy and accuracy.

7. The Collaborative Privacy Framework Prototype

We have implemented the collaborative privacy framework prototype with an aim to demonstrate the applicability of our approach in real life scenarios. However, we need to perform more design work in order to enhance its usability and make it friendlier to comply with the changing privacy practices and guidelines. The technologies used to develop our collaborative privacy framework are:

1. The proposed two stage concealment process is implemented in C++. The various local concealment algorithms were implemented using octave libraries. Moreover, the MPICH implementation of the MPI communication standard for distributed memory implementation of the global concealment algorithms to mimic a distributed reliable network of peers. To implement Paillier encryption scheme, the Number Theory Library (NTL) was used. One practical issue that must be dealt with when using the Paillier cryptosystem is the fact that it cannot naturally encrypt floating-point numbers. Floating-point numbers must be converted to a fixed-point representation. This is done by multiplying them by a large constant and then truncating the result to an integer.
2. The Aglets library was used to build different agents within the proposed *EMCP* middleware, which are running inside the user's device.
3. P3P policies and APPEL preferences rules standards were used to encode data collection, usage practices, and their actions.
4. MySQL database was used as data storage for storing users' profiles, polices, and privacy preferences that were acquired by the *EMCP* middleware.
5. Tor network was used to attain anonymity when sending data between different parties within the system, either between the participants and super-peers or between the super-peers and the social recommender service.
6. The experiments were conducted using the Jester and Moviedataset provided by Goldberg from UC Berkley [29] and Movielens dataset provided by Grouplens [30].

In order to set-up the proposed collaborative privacy framework, the users have to install the *EMCP* middleware on their personal devices (Setup box or mobile phone). Then after, they relocate their stored profiles into meta-data and ratings databases within the learning agent. Finally, they formalize their privacy preferences and actions for the various policies. The service provides are only required to offer P3P-compliant service by encoding their data collection and data usage practices in the form of P3P policies.

8. Conclusions and Future Work.

In this paper, we presented an attempt to develop an innovative approach for handling privacy in the current service oriented model. The collaborative privacy framework that was developed in complying with the OECD privacy principle has been depicted in detail. The proposed framework was implemented as a middleware that we have entitled *EMCP* "enhanced middleware for collaborative privacy". We gave a brief overview of *EMCP* architecture, components, and interaction sequence. We presented a novel two stage concealment process which provides complete privacy control to participants over their preferences. The concealment process utilizes a topological formation for data collection, where participants are organized into peer-groups, from which super-peers are elected based on their reputation. Super-peers and social recommender services use a platform for privacy preferences (P3P) policies for specifying their data usage practices. While participants describe their privacy constraints for the data extracted from their profiles in a dynamically updateable fashion using P3P policies exchange language (APPEL). The proposed framework allows a fine grained enforcement of privacy policies by allowing participants to ensure the extracted preferences for specific requests do not violate their privacy by automatically checking whether there is an APPEL preference corresponding to the given

P3P policy. Super-peers aggregate the preferences obtained from the underlying participants, encapsulate them in a group profile, and then send it to the social recommender service. We have tested the performance of the proposed framework on a case study for mobile jukebox recommender service using a real dataset. We evaluated how the overall accuracy of the recommendation varies based on various parameters of the two stage concealment process. The experimental and analysis results show that privacy increases under the proposed framework without hampering the accuracy of the recommendation. Thus, adding the proposed framework does not severely affect the accuracy of the recommendation based on the off-the-shelf recommendations techniques.

We realized that there would be many challenges in building a collaborative privacy framework for social recommender services. As a result, we focused on a middleware approach in our collaborative privacy solution. A future research agenda will include utilizing game theory to better formulate user groups, sequential preferences release and its impact on the privacy of the whole profile. Furthermore, it is included to strengthen our collaborative privacy framework against shilling attacks, extending our scheme to be directed towards multi-dimensional trust propagation and distributed collaborative filtering techniques in a P2P environment. We also need to perform extensive experiments on other real datasets from the UCI repository and compare our performance with other techniques proposed in the literature. Finally, we need to consider different data partitioning techniques as well as identify potential threats and add some protocols to ensure the privacy of the data against those threats.

9. Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education (2013R1A1A2061978)

10. References

[1] Olson, J.S., Grudin, J., and Horvitz, E.: ‘A study of preferences for sharing and privacy’. Proc. CHI '05 extended abstracts on Human factors in computing systems, Portland, OR, USA2005 pp. 1985-1988.

[2] Directive, E.: ‘95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data’, Official Journal of the EC, 1995, 23, pp. 6.

[3] Commission, E.: ‘Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector’, Official Journal L, 2002, 201, (31), pp. 07.

[4] Cranor, L.F.: ‘'I didn't buy it for myself' privacy and ecommerce personalization’. Proc. Proceedings of the 2003 ACM workshop on Privacy in the electronic society, Washington, DC2003 pp. 57-73.

[5] Cockcroft, S.K.S., and Clutterbuck, P.J.: ‘Attitudes towards information privacy’, Proc. Proceedings of the Twelfth Australasian Conference on Information Systems, Coffs Harbour, Australia, (School of Multimedia and Information Technology, Southern Cross University, 2001, edn.).

[6] Ramakrishnan, N., Keller, B.J., Mirza, B.J., Grama, A.Y., and Karypis, G.: ‘Privacy risks in recommender systems’, IEEE Internet Computing, 2001, 5, (6), pp. 54-63.

[7] Canny, J.: 'Collaborative Filtering with Privacy'. Proc. Proceedings of the 2002 IEEE Symposium on Security and Privacy2002 pp. 45-57.

[8] Canny, J.: 'Collaborative filtering with privacy via factor analysis'. Proc. Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, Tampere, Finland2002 pp. 238-245.

[9] Esma, A.: 'Experimental Demonstration of a Hybrid Privacy-Preserving Recommender System', Proc. Proceedings of the Third International Conference on Availability, Reliability and Security,Barcelona,Spain (2008, edn.), pp.161-170.

[10] Polat, H., and Du, W.: 'Privacy-Preserving Collaborative Filtering Using Randomized Perturbation Techniques'. Proc. Proceedings of the Third IEEE International Conference on Data Mining2003 pp. 625-628.

[11] Polat, H., and Du, W.: 'SVD-based collaborative filtering with privacy'. Proc. Proceedings of the 2005 ACM symposium on Applied computing, Santa Fe, New Mexico2005 pp. 791-795.

[12] Huang, Z., Du, W., and Chen, B.: 'Deriving private information from randomized data'. Proc. Proceedings of the 2005 ACM SIGMOD international conference on Management of data, Baltimore, Maryland2005 pp. 37-48.

[13] Kargupta, H., Datta, S., Wang, Q., and Sivakumar, K.: 'On the Privacy Preserving Properties of Random Data Perturbation Techniques'. Proc. Proceedings of the Third IEEE International Conference on Data Mining2003 pp. 99-106.

[14] Miller, B.N., Konstan, J.A., and Riedl, J.: 'PocketLens: Toward a personal recommender system', ACM Trans. Inf. Syst., 2004, 22, (3), pp. 437-476.

[15] Oliveira, S.R., and Zaïane, O.R.: 'Toward standardization in privacy-preserving data mining', Proc. Proceedings of the 3rd ACM SIGKDD Workshop on Data Mining Standards in conjuction with KDD 2004,Seattle, WA, USA, pp.7-17.

[16] Goecks, J., Edwards, W.K., and Mynatt, E.D.: 'Challenges in supporting end-user privacy and security management with social navigation'. Proc. Proceedings of the 5th Symposium on Usable Privacy and Security, Mountain View, California2009 pp. 1-12.

[17] Elmisery, A., and Botvich, D.: 'Enhanced Middleware for Collaborative Privacy in IPTV Recommender Services ', Journal of Convergence, 2011, 2, (2), pp. 33-42.

[18] Elmisery, A., and Botvich, D.: 'Privacy Aware Recommender Service using Multi-agent Middleware- an IPTV Network Scenario', Informatica, 2012, 36, (1), pp. 21-36.

[19] Elmisery, A., and Botvich, D.: 'Privacy Aware Recommender Service for IPTV Networks', Proc. Proceedings of the 5th FTRA/IEEE International Conference on Multimedia and Ubiquitous Engineering, 2011 Crete, Greece (IEEE, 2011, edn.), pp.160-166.

[20] Elmisery, A., and Botvich, D.: 'Private Recommendation Service For IPTV System: Protecting user profile privacy', Proc. Proceedings of the 2011 IFIP/IEEE International Symposium on Integrated Network Management (IM), Dublin, Ireland, (IEEE, 2011, edn.), pp. 571-577.

[21] Elmisery, A., Rho, S., and Botvich, D.: 'Collaborative Privacy Framework for Minimizing Privacy Risks in Social Recommender Services ', Submitted, 2014

[22] Elmisery, A., and Botvich, D.: 'Agent Based Middleware for Private Data Mashup in IPTV Recommender Services', Proc. Proceedings of the 16th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), Kyoto, Japan (IEEE, 2011, edn.), pp. 107-111.

[23] Kim, H.D.: 'Applying Consistency-Based Trust Definition to Collaborative Filtering', KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS, 2009, 3, (4), pp. 366-374.

[24] Parameswaran, R., and Blough, D.M.: 'Privacy preserving data obfuscation for inherently clustered data', Int. J. Inf. Comput. Secur., 2008, 2, (1), pp. 4-26.

[25] Dingledine, R., Mathewson, N., and Syverson, P.: 'Tor: the second-generation onion router'. Proc. Proceedings of the 13th conference on USENIX Security Symposium - Volume 13, San Diego, CA2004 pp. 303-320.

[26] Elmisery, A., and Huaiguo, F.: 'Privacy Preserving Distributed Learning Clustering Of HealthCare Data Using Cryptography Protocols', Proc. Proceedings of the 34th Annual Computer Software and Applications Conference Workshops (COMPSACW),Seoul,South Korea, (2010, edn.), pp. 140-145

[27] Elmisery, A., and Botvich, D.: 'Privacy Aware Obfuscation Middleware for Mobile Jukebox Recommender Services', Proc. Proceedings of the 11th IFIP WG 6.11 Conference on e-Business, e-Services, and e-Society, Kaunas, Lithuania, (IFIP, 2011, edn.), pp.73-86.

[28] Ghinita, G., Kalnis, P., and Skiadopoulos, S.: 'PRIVE: anonymous location-based queries in distributed mobile systems'. Proc. Proceedings of the 16th international conference on World Wide Web, Banff, Alberta, Canada2007 pp. 371-380.

[29] Gupta, D., Digiovanni, M., Narita, H., and Goldberg, K.: 'Jester 2.0 (poster abstract): evaluation of an new linear time collaborative filtering algorithm'. Proc. Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, Berkeley, California, United States1999 pp. 291-292.

[30] Lam, S., and Herlocker, J.: 'MovieLens Data Sets', in GroupLens Research Project, (Department of Computer Science and Engineering at the University of Minnesota., 2006, edn.).

Appendix D: Community Discovery & Recommendation Service Scenario

Article XI

Privacy Aware Community based Recommender Service for Conferences Attendees

Ahmed M. Elmisery, Kevin Doolin, Dmitri Botvich

In Proceedings of the 16th KES International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2012), San Sebastian, Spain, September 2012.

Advances in Knowledge-Based and Intelligent Information and Engineering Systems
M. Graña et al. (Eds.)
IOS Press, 2012

doi:10.3233/978-1-61499-105-2-519

Privacy Aware Community based Recommender Service for Conferences Attendees

Ahmed M. Elmisery, Kevin Doolin and Dmitri Botvich

Telecommunications Software & Systems Group-TSSG
Waterford Institute of Technology-WIT-Co. Waterford, Ireland

{ael-misery, kdoolin, dbotvich}@tssg.org

Abstract. With the rapid growth of social networks and users communities the need to attain privacy for end-users becomes mandatory especially with the recent privacy breaches and the inefficiency of anonymisation techniques [1]. The problem of maintaining privacy in recommender services become increasingly important since it aims at finding information that might be interesting to end-users without disclosing their real interests to the service. In this paper, we present a middleware that runs in end-users' mobile phones to provide referrals for joining different sub-communities in conferences or exhibitions in private way. Moreover, the proposed middleware facilitates identifying similarity between various attendees in order to build a community with specific interest without disclosing their real preferences or interests to other parties. Our proposed middleware equipped with two cryptography protocols in order to achieve this purpose. In such case, the attendees can submit their preferences in an encrypted form and the further computation of recommendation proceeds over the encrypted data using secure multi-party computation protocols. We also provide a scenario for community based recommender service for conferences along with experimentation results. Our results shows that our proposed middleware not only protect the attendees' privacy, but also can maintain the recommendation accuracy.

Keywords: Privacy; Clustering; Community Recommendations; Middleware

1 Introduction

With the increase of number of services available on the internet, there has been more demand for personalization services that can be used to fight against information overload and find information relevant to each user. Community based recommender service (CRS) aims at countering these problems by providing end-users referrals to join certain groups out of large number of communities that are relevant for a given end-user's interests. This service is based on the assumption that end-users with similar preferences have the same interests. CRS generates referrals based on end-user profiles containing, for each one, personal data and interests. The provided end-user profiles are usually structured and collected in two databases at the recommender service, namely end-user database and interests databases. CRS operates recommendations techniques on these data-

bases to determine users with the highest similarity between their interests in order to generate referrals for communities that are probably relevant for his/her profile.

From privacy point of view, many of the current recommender services have failed to meet the privacy requirements of end-users, which result in a lack of acceptance of the respective services in general. Privacy is an essential concern in all recommender services as generating referrals obviously requires the handling of private end-users' profiles. According to surveys results in [2], privacy aware users refrain from providing accurate information because of their fears of personal safety and the lack of laws that govern the use and distribution of these data. Based on another survey results in [3, 4] the end-users might leave a service provider because of privacy concerns. However, privacy concerns should be balanced with other general requirements regarding performance and accuracy of recommendations as well.

In this work, we present an enhanced middleware for collaborative privacy (*EMCP*) that allows creating reasonable referrals for joining various communities in a conference without breaching attendees' privacy. The participants' cooperation is needed not only to protect their privacy but also to allow the service to run properly. For the rest of this work, we will generically refer to attendees' preferences as interests. This paper is organized as follows. In Section 2, related works are described. Section 3 presents the threat model assumed in this work .Section 4 introduces private community based recommender service (PCRS) that is landing *EMCP*. The proposed protocols that are used in *EMCP* are introduced in details in Section 5. In Section 6, the Results from some experiments on the proposed mechanisms are reported. Finally, the conclusions and recommendations for future work are given in Section 7.

2 Related Works

The majority of the literature addresses the problem of privacy on social recommender services, due to it being a potential source of leakage of private information shared by the users as shown in [5]. In [6] a theoretical framework is proposed to preserve the privacy of customers and the commercial interests of merchants. Their system is a hybrid recommender system that uses secure two party protocols and public key infrastructure to achieve the desired goals. In [7, 8] a privacy preserving approach is proposed based on peer to peer techniques using users' communities, where the community will have a aggregate user profile representing the group as a whole but not individual users. Personal information is encrypted and communication done between individual users but not servers. Thus, the recommendations are generated on the client side. In [9] another method is suggested for privacy preserving on centralized recommender systems by adding uncertainty to the data by using a randomized perturbation technique while attempting to make sure that the necessary statistical aggregates such as the mean do not greatly get disturbed. Hence, the server has no knowledge about the true values of the individual items' ratings for each user. They demonstrate that this method does not essentially decrease the accuracy obtained in the results. But recent research work [10, 11] pointed out that these techniques do not provide levels of privacy as was previously thought. In [11] it is pointed out that arbitrary randomization is not safe because it is easy to breach the privacy protection it offers. Storing users' profiles on their own side and

running the recommender system in a distributed manner without relying on any server is another approach proposed in [12].

3 The Proposed Middleware

While various definitions of privacy exist in the literature, in the scope of this work, we want to introduce the notion of privacy within our solution in our terms. Privacy is the ability of an individual or group to seclude themselves or information about themselves and thereby reveal themselves selectively or based on levels. We seek to achieve privacy by implementing a privacy by design approach [13] where we consider a middleware that governs data collection and processing during community building process such that participants don't have to reveal plain interests in their profiles. This will help participants to control what they share with various communities and to join specific groups with a customized profile that access only to a subset of their data in their profile.

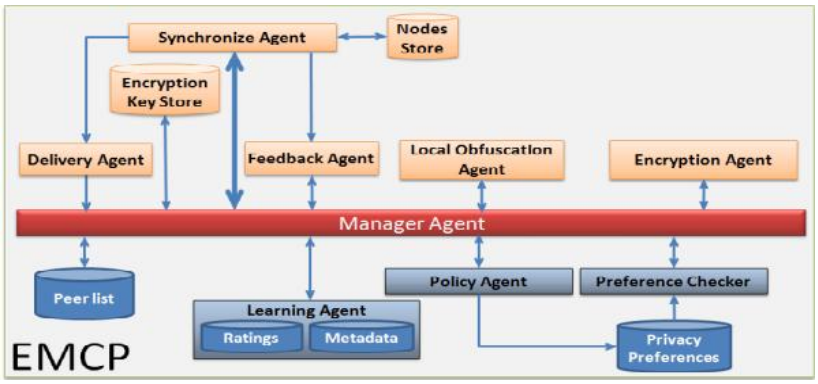


Fig. 1. *EMCP* Components

The intuition behind our solution stems from the fact that safest way to protect sensitive profiles data is to not publish them online, but keep them at user side. However, in order to gain most of PCRS's functionalities, attendees disclose their private data in some way to enable PCRS's functionalities. *EMCP* is implemented as a middleware running on top of attendees' mobile phones, a more precisely architecture of this middleware is presented in Figure (1). *EMCP* consists of different agents each of which has a certain task, but their co-operation is required to attain the whole functionality, more description about *EMCP* can be found in [14-18]. The local obfuscation agent creates a public profile that is used as an input to encryption agent. The encryption agent is responsible for executing two cryptographic protocols; first one is the private community formation (PCF) protocol which builds communities between attendees, while the other one is private sub-community discovery (PSD) protocol that help to discover sub-communities inside each community. These protocols act as wrappers that conceal interests before they are shared with any external entity. *EMCP* requires attendees to be organized into virtual topology which may be a simple ring topology or hierarchical topology, this ordering enables them to participate in multi-party computations as well. However, PCRS is the server that initiates the process to extract different communities and sub-communities. So, for any new attendee who wants to find a suitable community, he/she can submit his/her request to PCRS. The main steps for community building process are as following:

1. Each attendee are required to register himself /herself in a trusted third party called security authority center (SAC) prior using PCRS services, that will issue them anonymous credentials at the time of registration.
2. The attendees first submit a request to PCRS along with their anonymous credentials to start the process of forming communities.
3. PCRS accepts the request after authenticating the attendee's credentials.
4. PCRS will negotiate with each *EMCP* to select peers with highest reputations from each group that will serve as super-peers for collecting profiles from community's members. Then, PCRS submits this request to these super-peers.
5. The super-peers coordinate with their members to aggregate their hashed public profiles along with their anonymous credentials to create different communities.
6. For each community, participants encrypt their hashed private profiles then engage in peer to peer communication between other communities' members to discover the best sub-community that matches their expertise. Then they submit results to super-peer in order to form suitable sub-communities.
7. After forming communities and sub-communities, super-peers submit sub-communities representatives to both PCRS and community members.
8. At the end, each attendee obtains a list of sub-communities representatives and selects the one with highest similarity to his/her private profile.
9. For a new attendee who wants to join a sub-community, he/she contact PCRS and obtain list of sub-communities representatives then he/she encrypts his/her interests and then engages in matching protocol. This process will yield to one encrypted value for each community available that indicated the degree of similarity between community and attendee profile. *EMCP* assures that the PCRS cannot access the attendee's private profiles in clear form.

3.1 Threat Model

The proposed solution is secure in an honest-but-curious model. We assume that an adversary aims to collect attendees' interests in order to identify and track attendees. Thus, we consider our main adversary to be an untrusted CRS; moreover we do not assume CRS to be completely malicious. This is a realistic assumption because CRS needs to accomplish some business goals and increase its revenues. CRS can construct the profiles of the attendees based on the shared interests between sub-communities members. Hence, the problem we are tackling has two sides; we want to detain the ability of the adversary to identify attendee private interests based on a set of shared interests between sub-communities members. Intuitively, the system privacy is high if CRS is not able to reconstruct the real attendees' private interests.

3.2 Conference Communities Scenario

In this subsection, we present our scenario and analyze the participant interest privacy issues in community building process. Close inter-participant interactions have raised a new concern on the privacy of their interests; this is a key challenge in community building process due to the diversity and massive size of attendees-generated profiles. The scenario we are targeting here can be summarized as following based on conference

various themes, research strategies and specific topics the organizers propose different communities each of which has its own interaction space where any interactions are supported. Each attendee configures his *EMCP* to build a public profile that discloses some information about their general interests for networking and collaboration, In addition to contributing to the conference topics. Attendees seek to hide from the public their specific expertise, previous conference engagements, details of their research domains and problems in hand, current and previous funded projects, sessions and presentations they are planning to attend and finally their arrival/departure times. Other Private information such as names, company, etc, by default is protected by the privacy protection rules. In some cases where attendees already belonging to previously created group, they can form a sub-community inside the conference community they already belonging to such that they can participate in discussions and have access to the already exchanged opinions. PCRS provides information that can help conference organizers to determine numbers of attendees for each session, the linkages between attendees of similar interests to create opportunities for engagement, catering numbers and transport requirements. Moreover, the proposed system provides attendees with a personalized agenda and indoor navigation system to lead them to their desired sessions. *EMCP* can make use of all relevant information about the conference including access to session information, available services and other attendees' public profiles. The suitable topics and sessions are suggested for attendees based on their interests.

3.3 Cryptography Tools

Distributed Threshold Cryptosystem .

Using additively homomorphic cryptosystem permit the computation of linear combinations of encrypted data without need for prior decryption. Formally, an encryption schema $\varepsilon_{pk}(.)$ denotes the encryption function with encryption key pk and $D_{sk}(.)$ denotes the decryption function with decryption key sk. Paillier [19] proposed a probabilistic asymmetric algorithm for public key cryptography that is an example of an efficient additively homomorphic cryptosystem, this scheme is further extended by [20] with a threshold versions, but required the use of a trusted dealer to distribute the keys to the participants. The reliance on a trusted dealer was lifted in [21] to ensure that no single party or coalition of less than specific participants can recover the encrypted values. In designing our protocols, we require a fully distributed key generation protocol. In particular, the coalition between any parties within the group should not be able to decrypt the hashed private profiles. Therefore neither can be used as a trusted "dealer" for key generation. Moreover, it is desirable to distribute trust between numerous participants and no single party is assumed to be fully trusted. Thus, the decryption key sk is shared among a number P of participants, and encrypted profiles can only be decrypted only if any subset consisting of a threshold t of participants cooperate but no subset smaller than t can perform decryption.

Hash functions.

Using cryptographic hash function is an efficient one-way function that does not require the use of a secret key and is preimage resistant: given h, it is difficult to find any M such that $\mathrm{hash}(M) = h$. We will employ hash function to hide public interests while

exchange profiles data. To be more precise, an attendee v_b, which wants to compare his/her public profile with v_s's profile, and then he/she would simply hash all elements of his/her public profile, thus obtaining: $H(D_{v_s}) = \left\| \forall_{j \in L} \left(ID_j, hash(A_j^{v_s}) \right) \right\|$. As a counterpart, an intermediate participant v_c could compare his/her interests with v_s's interests in two steps, first v_c would first hash its profile with the same hash function hash. Then v_c tests whether one of its hashed interests hash $\left(A_j^{v_c}\right)$ is equal to an interest hash $\left(A_j^{v_s}\right)$ of the sender v_s. With help of the preimage resistance of hash, this implies that the interests where the equality holds are shared interests and the one where it does not hold are non-matching interests. Hash functions are efficient to compute thus it can be a public function in our middleware. Moreover, this idea is effective as the further computations of our protocols require only the size of interaction between various attendees.

4 Problem Formulation

In the following section we explain notions used in our solution, attendees' profiles can be represented in two categories public profiles $P(v)_{pub}$ and private profile $P(v)_{priv}$. Our goal is to protect private participants' profiles when formulating communities and recommending sub-communities since these are the information that attendees wish to keep private against both PCRS and third parties.

Definition 1. Our model is defined as follows: $v, i, level$ represents (participants, interests, access level) respectively. While permission $i \times level \rightarrow \{hypernym, Restrictive\}$ is a function that answers if an interest i can be published by *EMCP* in order to help him/her finding a suitable community.

Definition 2. (Synonym set) A synonym set of an interest is a set of words and phrases including the interest and all its synonyms. Synonym set introduces different lexical forms for attendees' interests. For example, (data mining, predictive analytics, statistical analysis) is the synonym set for any interest within this set.

Definition 3. (Hypernym path) In linguistics, a hypernym is a word or phrase whose semantic range includes that of another word, its hyponym. Hypernym path for a synonym set is a list of synonym sets including the root synonym set and all its hypernym sets. For example, data mining, predictive analytics, statistical analysis, are all hyponyms of machine learning (their hypernym).

Definition 4. The public profile of user v is a set of hypernym terms in the same semantic categories for the interests in attendee's profile. $P(v)_{pub} = \{i_i, i_2, \dots, i_n\} \forall\, i \in P(v)_{pub} \equiv$ user v assigns $i.level = hyperny$. The similarity between the terms on the public profile and the attendees' profile is computed then select the one that has the highest similarity.

Definition 5. The private profile of user v is a set of ests $P(v)_{priv} = \{i_i, i_2, \dots, i_m\} \forall\, i \in P(v)_{priv} \equiv i \notin P(v)_{pub}$ && $user\ v$ assigns $i.level =$ Restrictive||$\emptyset$.

In other words, we define $P(v)_{pub}$ as the generalized information that a user v configures his/her *EMCP* to disclose, while $P(v)_{priv}$ represents the "hidden" information that v does not want to disclose publically to others.

4.1 Proposed Privacy Enhanced Protocols for *EMCP*

We propose PCRS, a privacy aware community based recommender service for conference attendees. Privacy is attained using *EMCP* middleware which is hosted in attendees' mobile phones and equipped with cryptography protocols to perform secure multi-party computations for building communities. The PCRS consists of two key processes:

- Private community formation (PCF): in this process, *EMCP* executes PCF protocol that clusters attendees into general communities, such that each community contains various attendees who share the same interest. An attendee can belong to multiple communities.
- Private sub-community discovery (PSD): *EMCP* executes PSD protocol on the proximate general communities extracted from the first process, in order to determine in a bilateral manor the matched interests within attendee's private profiles to build sub-communities.

EMCP allows the formation of attendees' communities; such that attendees share the same experience can engage in discussions, exchange experiences or get contact of each other. Up to now, forming these communities relying on the chance of direct communicate between attendees to disseminate experiences. An important requirement for our solution for community formation is the ability of an attendee to search for and join a sub-community in private way. *EMCP* simplifies this process by breaking the correlation between attendees' public and private profiles to manage the various types of interests. The notion of community in this work can be defined:

Definition 6. A community is the set $C = \{c_1, c_2, \dots, c_n\}$, where n is the number of sub-communities in C, has the following properties: (1) Each $\forall_{i=1}^{n} c_i \in C$ is a 3-tuple $c = \{I_c, V_c, d_c\}$ such that $I_c = \{i_1, i_2, \dots, i_l\}$ is a set of interests, $V_c = \{v_1, v_2, \dots, v_k\}$ is a corresponding set of attendees, and $d_c \in I_c$ is the centroid of c. (2) For each tendee $\forall_{i=1}^{l} v_i \in V_c$, v have the interests V_c. (3) Centroid d_c has the smallest average distance from other preferences in V_c, and it represents the "core-point" of sub-community c. (4) For any two sub-communities c_a and c_b $(1 \leq a, b \leq n$ and $a \neq b)$, $V_{c_a} \cap V_{c_b} = \emptyset$ and $I_{c_a} \neq I_{c_b}$.

Private Community Formation (PCF) Protocol .

There are two main challenges in identifying communities from public profiles: the first one is representation of community, i.e., good intra-community similarity and inter-community separation. Then, second one is the protection of attendee' profile privacy in the process of community identification. We address these challenges as follows: Given a set of attendees V and a set of interests I, our goal is to cluster these attendees into k communities. In order to help attendees to build their public profiles, PCRS supply all the attendees with global information (e.g. concept taxonomy and term vocabulary) in-

dependently of their profile content, then attendees start mapping their profiles onto this global information space to get public profiles. After selecting super-peers, local obfuscation agent builds a public profile for each attendee using this global information in two steps:

1. Synonym set replacement. This step will replace each interest in attendee's profiles with its synonym set provided by PCRS.
2. Hypernym set. This step builds the hypernym sets for all existing synonym sets using concept taxonomy provided by PCRS.

Then participants submit a hashed version of their public profile to the elected super-peers. We will first consider the distance/similarity function between two attendees' profile data that can adequately capture the similarity of their interests and easy to calculate in a distributed fashion. Specifically, we leverage the Dice similarity for this task. Let $V_c(V_B)$ be the set of attendees who possess interests $I_A(I_B)$ then: $\text{UsersSimilarity}(V_{I_A}, V_{I_B}) = 2|V_{I_A} \cap V_{I_B}|/|V_{I_A}|^2 + |V_{I_B}|^2$. The set of interests that have high similarity will be clustered into the same community. We present in the next subsection the PCF protocol, which can compute the sum of k parties' private values without disclosing these values and partition participants to different communities. To utilize this protocol, we need to convert the set operations of $|V_{I_A} \cap V_{I_B}|$ and $|V_{I_A}|^2 + |V_{I_B}|^2$ into k party form. This can be achieved by defining each attendee's possession of specific interest as 1 (interested) or 0 (not interested). Given the PCF protocol, for a given s, the super-peer can then adopt the S-seeds clustering algorithm to cluster k participants into s different communities. PCF Protocol can be summarized as following:

- For any attendee $v \in V$ and a set of interests I, we denote v possession of an interest $i \in I$ as $P_{v,i} = 1$ and 0 otherwise. The dice similarity is calculated in two steps first, it computes the numerator $|V_{I_A} \cap V_{I_B}|$ a between each attendee v_A and v_B and then it computes the denominator $|V_{I_A}|^2 + |V_{I_B}|^2$.
- After selecting a super-peer as the root for computations, a ring topology for the attendees is employed for calculating the numerator between every two attendees. Each public profile is associated with certain interests that need to be compared with different attendees' public profiles then they submit similarity values to super-peers. Attendees who are willing to participate hash their interests $\{I_j\}_{1 \le j \le m}$ where each interest I_j defined with name N_j and value U_j. The attributes are distributed vertically, which means that the set $\{N_j\}_{1 \le j \le m}$ is known by all $\{V_i\}_{1 \le i \le p}$, the value of the attribute A_j at attendee v_i denoted by $U_{v_i j}$. For example super-peer A profile is composed of L interests' names required for comparison, so that $N^A_{L \subset [1,m]}$. When Attendee B ceives$A$'s profile it start comparing its own data D_B with N^A_L to extract the subset $\mathbb{Q} \subset L$ of values that are shared between A profile and D_B, such that $\forall\, n \in \mathbb{Q},\ N^A_n = (N^B_n, U_{Bn})$. After this subset extraction, B compute $|V_{I_B}|^2$ then it encrypt these values along with participants' pseudonyms identities using super-peer public key and forwards them with A's profile to the next participant in his group. Thus, participants are

able to correctly detect the matching hashed interests without revealing the non-matching ones.

- Super-peer collects all these results and decrypts them with its private key. Then it starts to cluster participants into communities. A participant can belong to multiple communities based on his/her public profile. Super-peer performs S-seeds clustering algorithm as following: first, it randomly select S attendees' profiles as clusters representatives. Then, it calculates the distance between these S seeds and each data point as specified in PCF protocol. Then, assign each point to the community with the closest seed. Inside each community, choose the point with the smallest average distance to other data as the new seed. Finally, repeat previous step until the S-seeds do not change. In S-seeds clustering, only the distance calculations among data points are required to identify the communities.

The above protocol performs it computations on m hashed values held by m parties without exposing any of the inputs values.

Private Sub-Community Discovery (PSD) Protocol .

The objective of this protocol is to compute sub-communities, i.e. each attendee determines its sub-community by considering only its private profile data and available communities that are obtained using PCF protocol. The list of available communities are shared between attendees and stored in PCRS. The attendees are arranged in hierarchical topology in order to compute sub-communities, PSD protocol can be summarized as follows:

- Each attendee A and B apply a locality sensitive hashing function lsh to encode their private profiles data and generate $E_A = lsh(D_A)$ and $E_B = lsh(D_B)$. In the same time, each attendee in sub-community engage in distributed key generation process using distributed threshold cryptosystem with other attendees to generate a complete public key PK along with a share of the private key SK.
- Attendee A encrypts independently his hashed data with PK $(E_A)_{PK}$ and sends it to other attendees in his community. For simplicity we assume that A sends his data to attendee B. Attendee B computes $(R_{AB})_{PK} = \prod_{i=1}^{n} \left((E_i^A)_{PK} (E_i^B)_{PK} ((E_i^A)_{PK})^{-E_i^B} \right)$. Furthermore, B select random value t and multiply it in the encryption $(tR_{AB})_{PK} = ((R_{AB})_{PK})^t$ then send the result to its next neighbor till the last attendee who submits the final result to the super-peer in the community.
- After super-peers receive encrypted values from their members, they start calculating the final similarity values for the members in the entire community without decrypting collected values, moreover decryption process requires number of participants to cooperates, since decryption done using their local share of the private key this make sure that no single party can get the profiles over a subset of participants in the community. Moreover, utilizing fully distributed threshold cryptosystem ensures that all collected profiles become useless after the termination of the process even if an attacker obtains the collected profiles. *EMCP* automatically destroys key shares directly after encrypting the hashed values. Super-peer then computes $|A \cap B|$ based on counting the number values $== (0)_{PK}$ then it calculate the similarity between sets A and B using Jaccard similarity $|A \cap B| / |A \cup B|$.Super-peer declares a match if similarity

value $\geq S$. Super-peers formulate a list of sub-communities based on both public profiles and similarity values, such that each sub-community has a center ε which represents the centroid vector for it.

- Super-peers submit the list of sub-communities representatives (ε, S) to PCRS in order to stores them along with public key that encrypts them. Moreover, super-peers publish these values to participants too.
- After each attendee receives (ε, S) values from super-peers, a final step of local recommendations generations needs to be performed locally at each attendee side. In this step, each attendee encrypts his private profile and then computes the distance between his profile and the values of each sub-community to generate personalized and ranked list of sub-communities that are mostly liked to him/her. this step separate attendees public profiles from private profiles, besides protecting their private interests it also ensures good recommendations quality.

5 Experiments

In this section, we describe the implementation of our proposed solution. The experiments are run on 2 Intel® machines connected on local network, the lead peer is Intel® Core i7 2.2 GHz with 8 GB Ram and the other is Intel® Core 2 Duo™ 2.4 GHz with 2 GB Ram. We used MySQL as data storage for the participants' profiles that is acquired by learning agent. PCRS has been implemented and deployed as a web service while *EMCP* has been deployed as middleware to handles the interactions between its owner, PCRS and other participants; it is implemented as an applet that uses implementation of the MPI communication standard for distributed memory implementation of our proposed protocols to mimic a distributed reliable network of peers. To implement Paillier encryption scheme, the Number Theory Library (NTL) was used. We tried PCRS to generate referrals for conference attendees to join various sub-communities, the experiments presented here were conducted using a dataset pulled from a recruiter network in Denmark (Manpower Professional) in period of 1990 t0 1997. It contains registration data and information related to different participants that attend exhibitions organized by this agent which held concurrently with various scientific conferences. This data set is comprised of approximately 67,000 users and contains various details about them. Each of those details fell into one of several categories: affiliation, expertise, domains, projects, activities, publication and awards. Due to the lack of a reliable subject authority, some other categories were discarded from all experiments. To generate the public profiles for affiliation, expertise, domains, projects, activities and awards categories we used Google Directories. To extract citations and conferences ranks for publications we used Google scholar then we classify citations into 3 categories [low, medium, high], moreover for publication's domain, we used the classification system for each research area like (ACM computing classification system). The experiments involve dividing the data set into a training set and testing set. The training set is used to create hashed public profiles that are used to generate communities then stored in the database of PCRS. Thereafter the original training dataset is hashed and encrypted and then presented to PCRS for extracting sub-communities from it. The testing dataset stands for new attendees who are looking for referrals to join different sub-communities. Our method employs secure mul-

tiparty computations using distributed threshold decryption (DTD) which requires cooperation between all community members to extract their original hashed private profiles. We evaluated the proposed solution from different aspects: privacy achieved, accuracy of results and performance. We used the famous precision and recall metrics to measure privacy and accuracy of the results. For the accuracy precision measures the portion of interests in sub-community s that attendee a likes $precision(s,a) = |P_a \cap P_s|/P_s$ while recall measures the portion of interests possessed by a that are actually in $recall(s,a) = |P_a \cap P_s|/P_a$. A Higher values for these metrics indicates better accuracy provided. While To measure the privacy or distortion achieved using our protocols, we will also use the previous metrics to measure the true positive interests that are inferred from attendee a private profile when he/she join a specific sub-community s, as these interests might be shared between all sub-community members. Based on this $precision(s,a) = \left|P_s^{shared} \cap P_a^{private}\right|/P_s^{shared}$ will measure the portion of interests that are shared by members and they are true private interests for attendee a and $recall(s,a) = \left|P_s^{shared} \cap P_a^{private}\right|/P_a^{private}$ refers to the portion of private interests possessed by a that are actually in these shared interests (privacy leak). A lower values for these metrics indicates a larger distortion between the shared and private interests, which means a higher level of privacy achieved. In the first experiment, we evaluated the recommendations accuracy of our proposed solution; the results are shown in figure (4). As we can see, a good quality is achieved due to:

1. Creating generalized communities in the start that involve various groups enable highly selective sub-communities recommendations to the attendees in this community. Since the community gathers participants who share the same general interests.
2. The effect on each interest inside the community can be easily measured, which enables to detect and remove outlier values that are very different than the general interests.

In the second experiments, we evaluated the leaked private interests of different attendees when running our solution. We consider users, who published portion of their real interests in their public profiles, for each of these users; we tried the attack procedure proposed in threat model to reveal other hidden interests in their profiles based on the community they belong. The obtained interests are quantified using our proposed metrics and the results are shown in figure (2). As we can see, our solution manages to reduce privacy leakages for exposed attendees' private interests. One important notice to put in consideration, our privacy metrics are pessimistic as the disclosed hashed interests agreed and published by the attendee in his/her public profile other interests are hashed hypernym terms for their private interests. The private profile is hashed and encrypted during the computation; moreover sub-community joining is determined at attendee side. Therefore, such information disclosure has a limited impact on private interests breach. On the other hand, sub-communities are represented with two values and collected attendees profiles are omitted from submission to PCRS.

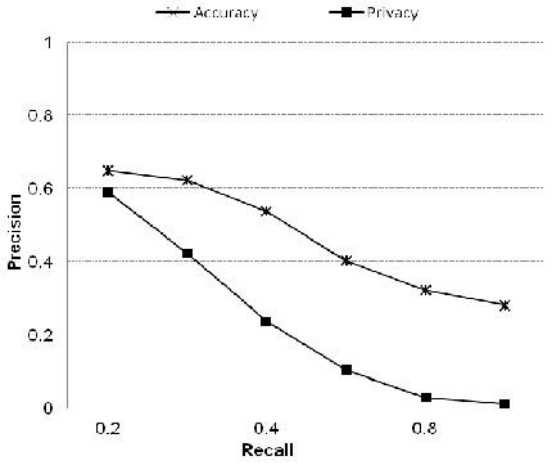


Fig. 2. Recommendations accuracy and privacy

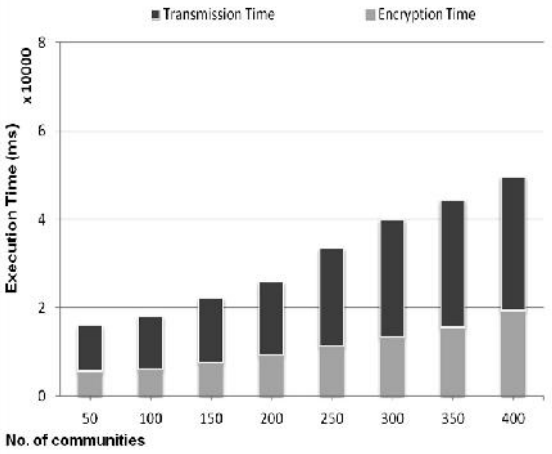


Fig. 3. Efficiency of our solution

In the last experiment, we want to measure efficiency of our solution with increasing number of communities. We measured execution time in terms of encryption and transmission time for participants' profiles, as we can see form Figure (3) our solution requires more communication in consequence of distributed design and communication needs for PCF and PSD protocols. This acceptable overhead is shared among all participants while the benefit is to protect their privacy without hampering recommendations quality. The processes of selecting super-peers and generating distributed key are done once in the setup time before the start of our protocols, so we omit the required time for them. Since super-peers are the bottleneck in our solution, we implement our distrusted computation using ring topology in order to reduce this risk and computation overhead.

6 CONCLUSION AND FUTURE WOK

In this paper, we presented our attempt to develop an enhanced middleware for collaborative privacy for community based recommender service in conferences or exhibitions. We gave a brief overview of *EMCP* architecture, components and recommendation process. We tested the performance of the proposed protovols on a real dataset. The experimental and analysis results show that privacy increases under the proposed middleware without hampering the accuracy of the recommendations. Thus adding the proposed middleware does not severely affect the accuracy of the recommendation techniques. A future research agenda will include utilizing game theory to better formulate user groups, sequential preferences release and its impact on privacy of whole profile. Finally, we need to consider reducing transmission time and the load on the network traffic.

ACKNOWLEDGMENT

This work partially supported by the European Comission via the ICT FP7 SOCIETIES Integrated Project (No. 257493). Also it was partially supported from the Higher Education Authority in Ireland under the PRTLI Cycle 4 Programme, in the FutureComm Project (Serving Society: Management of Future Communications Networks and Services).

References

[1] Narayanan, A., Shmatikov, V.: Robust De-anonymization of Large Sparse Datasets. Proceedings of the 2008 IEEE Symposium on Security and Privacy. IEEE Computer Society (2008)

[2] Teltzrow, M., Kobsa, A.: Impacts of user privacy preferences on personalized systems: a comparative study. Designing personalized user experiences in eCommerce. Kluwer Academic Publishers (2004)

[3] Cranor, L.F.: 'I didn't buy it for myself' privacy and ecommerce personalization. Proceedings of the 2003 ACM workshop on Privacy in the electronic society. ACM, Washington, DC (2003)

[4] Dialogue, C.: Cyber Dialogue Survey Data Reveals Lost Revenue for Retailers Due to Widespread Consumer Privacy Concerns. Cyber Dialogue (2001)

[5] McSherry, F., Mironov, I.: Differentially private recommender systems: building privacy into the net. Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, Paris, France (2009) 627-636

[6] Esma, A.: Experimental Demonstration of a Hybrid Privacy-Preserving Recommender System. In: Gilles, B., Jose, M.F., Flavien Serge Mani, O., Zbigniew, R. (eds.), Vol. 0 (2008) 161-170

[7] Canny, J.: Collaborative filtering with privacy via factor analysis. Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, Tampere, Finland (2002) 238-245

[8] Canny, J.: Collaborative Filtering with Privacy. Proceedings of the 2002 IEEE Symposium on Security and Privacy. IEEE Computer Society (2002) 45

[9] Polat, H., Du, W.: SVD-based collaborative filtering with privacy. Proceedings of the 2005 ACM symposium on Applied computing. ACM, Santa Fe, New Mexico (2005) 791-795

[10] Huang, Z., Du, W., Chen, B.: Deriving private information from randomized data. Proceedings of the 2005 ACM SIGMOD international conference on Management of data. ACM, Baltimore, Maryland .

[11] Kargupta, H., Datta, S., Wang, Q., Sivakumar, K.: On the Privacy Preserving Properties of Random Data Perturbation Techniques. Proceedings of the Third IEEE International Conference on Data Mining. IEEE Computer Society (2003) 99

[12] Miller, B.N., Konstan, J.A., Riedl, J.: PocketLens: Toward a personal recommender system. ACM Trans. Inf. Syst. 22 (2004) 437-476

[13] Rubinstein, I.: Regulating Privacy by Design. Berkeley Technology Law Journal, Forthcoming (2011)

[14] Elmisery, A., Botvich, D.: Privacy Aware Recommender Service using Multi-agent Middleware- an IPTV Network Scenario. Informatica 36 (2012)

[15] Elmisery, A., Botvich, D.: Enhanced Middleware for Collaborative Privacy in IPTV Recommender Services Journal of Convergence 2 (2011) 10

[16] Elmisery, A., Botvich, D.: Agent Based Middleware for Maintaining User Privacy in IPTV Recommender Services. 3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems. ICST, Aalborg, Denmark (2011)

[17] Elmisery, A., Botvich, D.: An Agent Based Middleware for Privacy Aware Recommender Systems in IPTV Networks. 3rd International Conference on Intelligent Decision Technologies Springer Verlag, University of Piraeus, Greece (2011)

[18] Elmisery, A., Botvich, D.: Privacy Aware Recommender Service for IPTV Networks. 5th FTRA/IEEE International Conference on Multimedia and Ubiquitous Engineering. IEEE, Crete, Greece (2011)

[19] Paillier, P.: Public-Key Cryptosystems Based on Composite Degree Residuosity Classes.

[20] Damgård, I., Jurik, M.: A Generalisation, a Simpli.cation and Some Applications of Paillier's Probabilistic Public-Key System Public Key Cryptography. In: Kim, K. (ed.), Vol. 1992. Springer Berlin Heidelberg (2001) 119-136

[21] Damgård, I., Koprowski, M.: Practical Threshold RSA Signatures without a Trusted Dealer Advances in Cryptology — EUROCRYPT 2001. In: Pfitzmann, B. (ed.), Vol. 2045. Springer Berlin / Heidelberg (2001) 152-165

Appendix D: Community Discovery & Recommendation Service Scenario

Article XII

Enhanced Middleware for Collaborative Privacy in Community based Recommendations Services

Ahmed M. Elmisery, Kevin Doolin, Ioanna Roussaki, Dmitri Botvich

In Proceedings of the 4th FTRA International Conference on Computer Science and its Applications (CSA 12), Jeju, Korea, November 2012.

Enhanced Middleware for Collaborative Privacy in Community based Recommendations Services

Ahmed M. Elmisery [1], Kevin Doolin [1], Ioanna Roussaki [2] and Dmitri Botvich [1]

[1] TSSG, Waterford Institute of Technology-WIT-Co. Waterford, Ireland
[2] National Technical University of Athens, Athens, Greece

Abstract. Recommending communities in social networks is the problem of detecting, for each member, its membership to one of more communities of other members, where members in each community share some relevant features which guaranteeing that the community as a whole satisfies some desired properties of similarity. As a result, forming these communities requires the availability of personal data from different participants. This is a requirement not only for these services but also the landscape of the Web 2.0 itself with all its versatile services heavily relies on the disclosure of private user information. As the more service providers collect personal data about their customers, the growing privacy threats pose for their patrons. Addressing end-user concerns privacy-enhancing techniques (PETs) have emerged to enable them to improve the control over their personal data. In this paper, we introduce a collaborative privacy middleware (*EMCP*) that runs in attendees' mobile phones and allows exchanging of their information in order to facilities recommending and creating communities without disclosing their preferences to other parties. We also provide a scenario for community based recommender service for conferences and experimentation results.

Keywords: Privacy; Clustering; Community Recommendations; Middleware

1 Introduction

With the popularity of social networks in the last few years, users are incited to build profiles containing their preferences, join different groups and utilize various services provided within the social platform. Community based recommender service (CRS) is a service running on social media platform and aims at providing end-users referrals to join certain sub-communities out of large number of communities that are relevant for a given end-user's interests. This service is based on the assumption that end-users with similar preferences have the same interests. CRS generates referrals based on end-user profiles containing, for each one, personal data and interests. The CRS is usually accessible and open to all attendees. However, this flexibility brings forward new threats and problems such as malicious behaviors against different participants from both service provider and other participants. For instance, malicious users may get one another's private information, such as current and previous occupations, age and relationship status, even if for the user the information is not supposed to be exposed publicly.

Several strategies have been proposed to control the disclosure of private information. The most popular approach is to permit users to maintain a set of privacy rules, according to which a decision is performed whether to release or not certain preferences in owner profile. However, these approaches are either rather coarse-grained, or require a deep understanding of the privacy

control system, any change of one privacy setting may result in unwanted or unexpected behaviors. Moreover, these approaches are based on the logic of either to allow or deny releasing certain preferences in users' profiles. Once, the data is released the user have no control over it and users will be vulnerable for the privacy breaches since released pieces of users' information is often interleaved, adversaries may be able to infer other private information using inference techniques. For example work in [1] shows that private information can be inferred via social relations, and the stronger the relationships people have in the network, the higher inference accuracy can be achieved.

In this paper, we lay out recommending and creating communities functions within user-side, this privacy architecture will help foster the usage and acceptance of our proposed protocols and eliminates the risk of possible privacy abuses as the sensitive data is only available to the owner but not to any other parties. However, as a consequence of applying our protocols, the structure in data is destroyed. In order to facilitate processing of such data, our protocols maintain some properties in this data which is suitable for the required computation. In rest of this work, we will generically refer to attendees' preferences as interests. This paper is organized as follows. In Section 2, related works are described. Section 3 presents the proposed middleware *EMCP* used in this work .Section 4 introduces some definition required for this paper. The proposed protocols that are used in *EMCP* are introduced in details in Section 5. In Section 6, the Results from some experiments on the proposed mechanisms are reported. Finally, the conclusions and recommendations for future work are given in Section 7.

2 Related Works

The majority of the literature addresses the problem of privacy on social recommender services, due to it being a potential source of leakage of private information shared by the users as shown in [2]. In [3] a theoretical framework is proposed to preserve the privacy of customers and the commercial interests of merchants. Their system is a hybrid recommender system that uses secure two party protocols and public key infrastructure to achieve the desired goals. In [4, 5] a privacy preserving approach is proposed based on peer to peer techniques using users' communities, where the community will have a aggregate user profile representing the group as a whole but not individual users. Personal information is encrypted and communication done between individual users but not servers. Thus, the recommendations are generated on the client side. Storing users' profiles on their own side and running the recommender system in a distributed manner without relying on any server is another approach proposed in [6].

3 The Proposed Middleware

In the scope of this work, we aim to achieve privacy by empowering an individual or group to seclude themselves or information about themselves thereby reveal themselves selectively or based on levels. We seek to achieve privacy by implementing a privacy by design approach [7] where we consider a middleware that governs data collection and processing during community building process such that attendees don't have to reveal private interests in their profiles. This will help them to control what they share with various communities and to join specific sub-community with a customized profile that access only to a subset of their interests. The intuition behind our solution stems from the fact that safest way to protect sensitive profiles data is to not

publish them online, but keep them at user side. However, in order to gain most of PCRS's functionalities, attendees disclose their private data in some way to enable PCRS's functionalities.

EMCP (enhanced middleware for collaborative privacy) is implemented as a middleware running on top of attendees' mobile phones [8-13]. *EMCP* consists of different agents each of which has a certain task, but their co-operation is required to attain the whole functionality. The local obfuscation agent creates a public profile that is used as an input to encryption agent. The encryption agent is responsible for executing two cryptographic protocols; first one is private community formation (PCF) protocol which builds general communities based on attendees' profiles, while the other one is private sub-community discovery (PSD) protocol that help to discover sub-communities inside each community. These protocols act as wrappers that conceal interests before they are shared with any external entity. *EMCP* requires attendees to be organized into virtual topology which may be a simple ring topology or hierarchical topology, this ordering enables them to participate in multi-party computations as well. However, PCRS (private community based recommender service) is the server that initiates the process to extract different communities and sub-communities. The scenario we are considering here is the one introduced in [8] it can be summarized as following based on conference various themes, research strategies and specific topics, the organizers setup a list of available communities on PCRS which act as interaction space that supports any interactions between attendees. Each attendee configures his *EMCP* to build a public profile that discloses some information about their general interests that are related to conference topics for the purpose of networking and collaboration. Attendees seek to hide from the public their specific expertise, previous conference engagements, details of their research domains and problems in hand, current and previous funded projects, sessions and presentations they are planning to attend and finally their arrival/departure times. Other Private information such as names, company, etc, by default is protected by the privacy protection laws. If attendees already belonging to previously created group, they can form a sub-community inside the conference community such that they can participate in discussions and have access to the already exchanged opinions. *EMCP* provides referrals to suitable sub-communities and sessions for attendees based on their interests.

3.1 Threat Model

The proposed solution is secure in an honest-but-curious model. Where, every party is obliged to follow the protocol but they are curious to find out as much as possible about the other inputs. The adversaries we consider here are untrusted CRS and malicious attendees that aim to collect other attendees' interests in order to identify and track them. Moreover we do not assume CRS to be completely malicious. This is a realistic assumption because CRS needs to accomplish some business goals and increase its revenues. Intuitively, the system privacy is high if CRS is not able to reconstruct the real attendees' private interests.

4 Problem Formulation

In the following section we outline important notions used in our previous solution in [8] and required in this work, attendees' profiles can be represented in two categories public profiles and private profile. Public profiles is a set of hypernym terms in the same semantic categories for the interests in attendee's profile [8], it represent general information that attendee configures his/her

EMCP to disclose, while private profile represents the "hidden" interests that attendee does not want to disclose publically to others. Our goal is to protect private participants' profiles when formulating communities and recommending sub-communities since these are the information that attendees wish to keep private against both PCRS and third parties. The notion of community in this work can be defined:

Definition 1. A community is the set $C = \{c_1, c_2, \dots, c_n\}$, where n is the number of sub-communities in C, has the following properties: (1) Each $\forall_{i=1}^{n} c_i \in C$ is a 3-tuple $c = \{I_c, V_c, d_c\}$ such that $I_c = \{i_1, i_2, \dots, i_l\}$ is a set of generalized interests, $V_c = \{v_1, v_2, \dots, v_k\}$ is a corresponding set of attendees, and $d_c \in I_c$ is the main-interest of c. (2) For each attendee $\forall_{i=1}^{l} v_i \in V_c$, v have the interests V_c. (3) d_c is the frequent interest in V_c profiles, and it represents the "core-point" of sub-community c. (4) For any two sub-communities c_a and c_b $(1 \leq a, b \leq n \text{ and } a \neq b)$, $V_{c_a} \cap V_{c_b} = \emptyset$ and $I_{c_a} \neq I_{c_b}$.

5 Proposed Privacy Enhanced Protocols for *EMCP*

In our architecture, privacy is attained using *EMCP* middleware which is hosted in attendees' mobile phones and equipped with two cryptography protocols which are private community formation protocol (PCF) and private sub-community discovery protocol (PSD) that build communities and sub-communities. *EMCP* allows the formation of attendees' communities; such that attendees share the same experience can engage in discussions and exchange experiences. An important requirement for our solution is the ability of an attendee to search for and join various sub-communities in private way.

5.1 Private Community Formation (PCF) Protocol

Our aim is to cluster attendees' profiles into different communities. There are two challenges in identifying these communities: first one is representation of community, i.e., good intra-community similarity and inter- community separation. And the second one is the protection of private profiles in the process of community identification. In order to do so, attendees build public profiles using global information supplied by PCRS (e.g. concept taxonomy and term vocabulary) independently of their profile content, then local obfuscation agent at attendees side start mapping their profiles into this global information space to get public profiles as proposed in [8]

After building public profiles, *EMCP* invokes the encryption agent to execute PCF protocol that is responsible for clustering attendees into general communities, such that each general community contains various attendees who share similar interests in their profiles. An attendee can belong to multiple communities, thus allowing the separation between public profiles from his/her private profiles. Our novel secure multi-party computation protocol ensures participants privacy when forming communities and matching participant public profile with the list of available communities. PCF is executed in distrusted manor; it first creates a bag of interests representations of each attendee using their profiles data. Then, the extracted interests (words) are stemmed and filtered using domain-specific dictionary; these interests associated with a user V_c are used to create a word vector $V_c = (e_c(w_1), \dots . e_c(w_m))$, where m is the total number of distinct words in his/her is profile, and $e_c(w_1)$ describes the degree of importance of user V_c in interest w_1 (weighted frequency). The further computation proceeds to calculate term frequency inverse profile frequency [14] as following :

$$Term - frequency_{V_c}(w_i) = \#w_i \; in \; V_c \; profile / \#words \; in \; V_c \; profile$$
$$inverse - profile - frequency_{V_c}(w_i) = log(\#user / \#profiles \; contain \; word \; w_i)$$
$$e_c(w_1) = Term - frequency_{V_c}(w_i) * inverse - profile - frequency_{V_c}(w_i)$$

The similarity function between two attendees' profiles data should adequately capture the similarity of attendees' interests, and should be easy to calculate in a distributed and private fashion. Specifically, we leverage the Dice similarity for this task. Let $V_c(V_d)$ be the two word vectors for attendees C and D then:

$$UsersSimilarity(V_c, V_d) = 2|V_c \cap V_d| / |V_c|^2 + |V_d|^2$$

Intuitively, this means that two attendees C and D would be considered similar if they share many common words in their associated profiles, and even more so if only a few users share those words. Users have high similarity in set of interests will be clustered into the same community. To protect user privacy, an attendee's interests are stored locally and are not disclosed to other parties including the PCRS. Therefore, a secure multi-party computation mechanism is needed to compute the similarity between every two attendees. We present in the next sub-section the similarity calculation procedure in PCF protocol as follows:

1. For any attendee $C, D \in V$ and a set of word vectors $e_c(w_i)$ and $e_d(w_i)$, the similarity is calculated in two steps first, it computes the numerator $|V_C \cap V_D|$ between attendee C and D and then it computes the denominator$|V_C|^2 + |V_D|^2$.
2. After selecting a super-peer as the root for computations, a virtual ring topology between attendees is employed for calculating the numerator between every two participants. Each public profile is associated with certain interests that need to be compared with other participants' public profiles then they submit similarity values to super-peers. Both attendees C and D apply a hash function h to each of their word vectors to generate $V_c = h(\, e_c(w_i))$ and $V_d = h(\, e_d(w_i))$. *EMCP* at attendee C generates an encryption E and decryption U keys then it submit the encryption key E to D.
3. Encryption agent at attendee D hides V_d by $B_d = \{e_d(w_i) \times r^D | w_i \in V_d\}$ where r is a random number for each interest w_i, and send B_d to C.
4. Encryption agent at attendee C signs B_d and get the signature S_d, then sends S_d to D again with the same order it receives. *EMCP* at attendee D reveals set S_d using the set of r values and obtains the real signature SI_d, then it applies hash function h on SI_d to produce $SIH_d = H(SI_d)$.
5. Encryption agent at attendee C signs the set V_c and gets signature SI_c then applies same hash function h on SI_d to produce $SIH_c = H(SI_c)$ and submits this set to D.
6. Encryption agent at attendee D compares SIH_d and SIH_c using the knowledge of V_d, D gets the intersection set $IN_{C,D} = SIH_c \cap SIH_d$ that represent$|V_c \cap V_D|$. *EMCP* at D applies hash function h on $IN_{C,D}$ then it encrypts this value along with $|V_D|$, $|V_C|$ and attendees pseudonyms identities using super-peer public key and forwards them to super-peer of this group.
7. Super-peer collects all these results and decrypts them with its private key. Then it starts to cluster participants into communities, such that each community contains participants who share similar interests. Super-peer performs S-seeds [8] clustering algorithm as follows first, randomly select S attendees' profiles as clusters representatives. Then, it calculates the distance between these S seeds and each data point as specified in PCF protocol. Then, assigns each point to the community with the closest seed. Inside each community, choose the point with the smallest average distance to other data as the new seed. Finally, repeat last two steps until the S-seeds do not change. In S-seeds clustering, only the distance calculations among data points are required to identify the communities without disclosing attendees' profiles.

The above protocol performs it computations on m hashed values held by m parties without exposing any of the inputs values. This protocol is based on secure multi-party computation (SMPC), which was studied first by Yao in his famous Yao's millionaire problem [15].

5.2 Private Sub-Community Discovery (PSD) Protocol

Encryption agent in *EMCP* executes PSD protocol on the proximate general communities extracted from PCF protocol, PSD protocol determines in a bilateral manor the associated interests within attendees' public profiles, then the final results is used in building sub-communities. PSD protocol is adapted from the work in [16, 17] with the intuition that many frequent interests of attendees should be shared within a sub-community (group) while different sub-communities should have more or less different frequent interests. However, there are no predefined sub-communities yet inside these communities; hence PSD should operate with the available bounded prior domain knowledge and full dimensional profiles

Definition 2. (Frequent interests) Frequent interests is a notion similar to frequent itemsets in association rule mining, it represent a set of interests that occur together in some minimum fraction of attendees' profiles. For example, let's consider two frequent interests, "libraries" and "C". Profiles containing the interest "libraries" may relate to digital archiving services and Profiles that contain the interest "C" may relate to Healthcare services. However, if both interests occur together in many profiles, then a specific interest sub-community related to C-programming should be identified.

Definition 3. (Global Frequent Interests) Global frequent interests is a set of interests that appear together in more than a minimum fraction of the whole attendees 'profiles in community C; a minimum community support is specified for this purpose. If this set contains k-interests, it called global frequent k-interests such that each interest that belongs to this set is called global frequent interest. Global frequent interest is frequent in sub-community c_i if this interest is contained in some minimum fraction of attendees' profiles; a minimum sub-community support is specified for this purpose.

The attendees are arranged in hierarchical topology in order to compute sub-communities, PSD protocol can be summarized as follows:

1. The initialization process of PSD protocol is invoked by PCRS, whereas attendees form groups then after they negotiate with each other to elect a peer who will act as a "super-peer" for each group. Super-peers distribute a list of 1-candidate frequent interests; therefore, different group members run concurrently a local algorithm to generate local frequent interests using their local support and closure parameters. we use the algorithm presented in [18] to find global & local frequent interests for each group.
2. After local extraction of frequent interests at each member $\forall_1^n \mathrm{P_i}$, member $\mathrm{P_i}$ encrypts this local list with his own key and send it to member $\mathrm{P_{i+1}}$, such that each member successively sends both his local and received lists to next neighbor. Last member in the group $\mathrm{P_{n-1}}$ send collected message to the super-peer. Super-peers now, have a set of local supports and closures of candidate frequent interests; generating global support is done by making the sum of these local supports. The global closure is calculated using intersection of the collected local closure. In the same way, repeating the previous steps, super-peer can generate the candidates of higher size. In order to decrypt the final results, the super-peer encrypts and sends global supports & closures lists to member $\mathrm{P_{n-1}}$ in arbitrary order. Member $\mathrm{P_{n-1}}$ decrypts his encryption from

these lists using his own private key, and then sends this list to the next member P_{n-2} in arbitrary order. When super-peer receives these lists back, these lists will be encrypted with his own key only, which enables him/her to obtain final results.

3. For each adjacent set of global frequent interests at super-peer side, we setup an initial sub-community that includes all attendees' profiles that contain these interests, such that all profiles in this sub-community contain all these global frequent interests. These initial sub-communities are overlapped because each profile may contain multiple global frequent interests. PSD will use these global frequent interests as a sub-community representative. Then after, for each attendee's profile V_i , encryption agent determines the best initial sub-community c_i using the following score function: $SimilarityScore(c_i \leftarrow V_i) = \left[\sum_{w_i} e_r(w_i) * sub-community\ support(w_i)\ \right] - \left[\sum_{w_i'} e_r(w_i') * community\ support(w_i')\right]$
   Where w_i is a global frequent interest in profile r and this interest is also frequent in sub-community c_i while w_i' is a global frequent interest in profile r and is not frequent in sub-community c_i. $e_r(w_i)$ and $e_r(w_i')$ are the weighted frequency of w_i and w_i' in profile r, which already calculated during the execution of PCF protocol. After this scoring, each attendee's profile belongs to exactly one sub-community.
4. For each community, super-peer organizes sub-communities in hierarchical structure using global frequent k-interests in each sub-community as representatives. In that case, PSD treats all attendees' profiles in each sub-community as single conceptual profile. The sub-community with k-interests will appear at level k in this structure, while the parent sub-community at level k-1 must be a subset of its child sub-community's representatives at level k. The selection of the potential parent for each child sub-community is done using scoring function presented in previous step. After that, super-peers exchange discovered sub-communities with each other to efficiently remove the overly sub-communities based on inter sub-community similarity. The same frequent interests might be distributed over multiple small sub-communities obtained from different super-peers' results, thus merging every two sub-communities into one general sub-community occurs only if they are very similar to each other. Inter sub-community similarity is similar to scoring function presented before with the only difference is that this similarity value should be normalized to remove the effect of varying number of attendees in each sub-community, it is measured using the following functions:

$$SubcommunitySimilarity(c_i \leftarrow c_j) = \left[SimilarityScore\left(c_i \leftarrow \forall_{x=1}^{n} V_x \in c_j\right) \Big/ \left[\sum_{w_j} e(w_j) + \sum_{w_j'} e(w_j')\right] \right] + 1$$

   Then, $Inter\ subcommunity\ similarity\ (c_i \leftrightarrow c_j) = \left[SubcommunitySimilarity(c_i \leftarrow c_j) * SubcommunitySimilarity(c_j \leftarrow c_i)\right]$
   Where c_i and c_j are two sub-communities; $\forall_{x=1}^{n} V_x \in c_j$ stands for single conceptual profile for sub-community c_j. w_j represents a global frequent interest in both c_i and c_j while w_j' represent a global frequent interest in c_j only but not in c_i. $e(w_j)$ and $e(w_j')$ are the weighted frequency of w_j and w_j' sub-community c_j.
5. Finally, for a new attendee, in order to privately recommend suitable sub-communities for him/her, *EMCP* obtains a list of sub-communities representatives then it generalizes his/her host interests and extract frequent interests for this generalized profile. *EMCP* encrypts these

frequent interests and measure their similarity with sub-communities' representatives in order to build a list of similar sub-communities. Finally *EMCP* assigns his/her host to the sub-community with the highest similarity.

6 Experiments

In this section, we describe the implementation of our proposed solution. The experiments are run on 2 Intel® machines connected on local network, the lead peer is Intel® Core i7 2.2 GHz with 8 GB Ram and the other is Intel® Core 2 Duo™ 2.4 GHz with 2 GB Ram. We used MySQL as data storage for the participants' profiles that is acquired by learning agent. PCRS has been implemented and deployed as a web service while *EMCP* has been deployed as an applet to handles the interactions between its owner, PCRS and other participants; it uses the implementation of the MPI communication standard for distributed memory implementation of our proposed protocols to mimic a distributed reliable network of peers. Our proposed protocols implemented using Java and boundycastel© library, RSA key length is set to 512 for the experimental scenario. The experiments were conducted using a dataset pulled from a recruiter network in Denmark (Manpower Professional) in period of 1990 t0 1997. It contains registration data and information related to different participants that attend exhibitions organized by this agent which held concurrently with various scientific conferences. This data set is comprised of approximately 67,000 users and contains various details about them. Each of those details fell into one of several categories: affiliation, expertise, domains, projects, activities, publication and awards, etc. Due to the lack of a reliable subject authority, some other categories were discarded from all experiments. To generate the public profiles from these profiles we use same method proposed in [8].

In the first experiment, we want to measure the execution time for PCF protocol, from first step to last step at each attendee (excluding the time required to generate RSA keys). We divided our dataset into approximately same number of records and distribute then between 20 participants, then we run this experiment 7 times, so each point in the Fig. (1) is the mean value of the 7 runs. Additionally, we performed two other experiments in our dataset in which data was not divided into parts of same number of records. The first experiment, one client got 60% of total number of records and the rest of records were divided to other clients as parts of approximately same number of records. While, in the second one, one client got 40% of total number of records, other clients got the rest. The results of these experiments are summarized in Fig. (1). The results indicate the performance benefits of our protocol, as it is not sensitive to the number of shared interests.

In the next experiment, we need to measure the accuracy of extracted sub-communities using PSD protocol. In order to evaluate the accuracy of our results, we apply hierarchical agglomerative clustering in our dataset in order to indentify natural sub-communities from attendees' private profiles. These sub-communities are utilized for measuring the accuracy of the results produced by PSD protocol. Each cluster represents a sub-community which is constructed from a set of attendees' private profiles who share the same specific interests about the same topic. To measure the goodness of our results, we considered two error metrics defined in [19] which are grouping error (GR) and critical error (CIE). The first one, the grouping error (GR), takes into account the number of attendees' profiles included in a sub-community, but belonging to a topic different from the dominant topic in that sub-community. The second one, the critical error (CIR) measures the number of attendees' profiles belonging to a topic that is not the dominant one in any sub-

community. The graphs in Fig.(3) and (4), contain both GR and CIE values for the results obtained from both hierarchical clustering and PSD protocol for different number of sub-communities. This experiment is performed on two versions of our dataset; attendees' generalized profiles are utilized by our PSD protocol, while hierarchical agglomerative clustering utilizes attendees' private profiles that should kept private in our scenario.

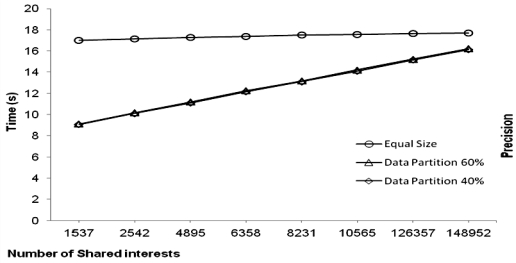


Fig.1. Execution Time for PCF Protocol

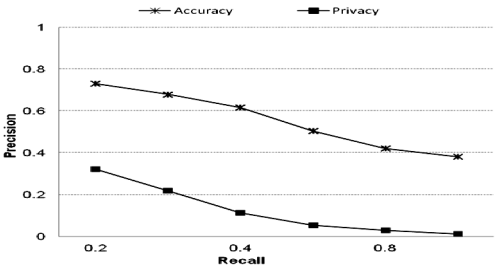


Fig.2. Recommendations Accuracy and Privacy

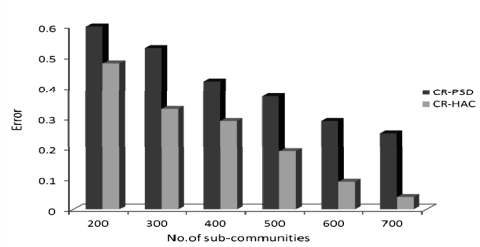


Fig.3. Grouping Error (GR) of PSD Protocol

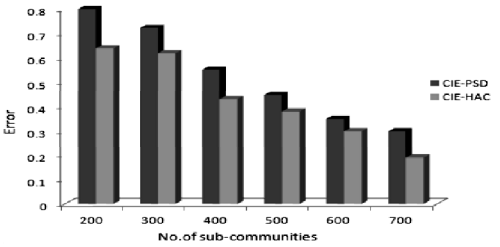


Fig. 4. Critical Error (CIE) of PSD Protocol

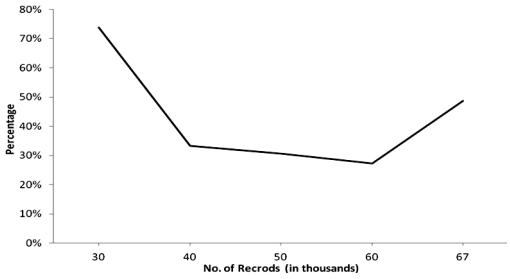


Fig. 5. Percentage Time for PSD Protocol

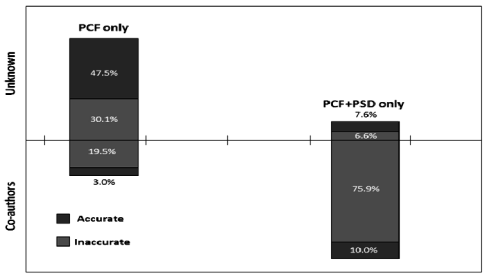


Fig.6. Co-authors vs. Unknown, Accurate vs. Inaccurate

We can deduce that both GR and CIE for PSD decrease with the increase in no. of sub-communities till reaching natural number of sub-communities. This indicates that achieving privacy is feasible and does not severely affect the accuracy of the generated sub-communities.

In the last experiment on PSD protocol, we want to measure the overhead of the execution time when applying PSD protocol to preserve attendees' privacy. We divided our dataset into different number of records from 30.000 to 67.000, such that each party held approximately the same number of records. We recorded the execution time when applying our PSD with encryption and without encryption on this data, then we calculated the percentage as following: $percentage = \left({}^{time\ without\ encryption}/_{time\ with\ encryption}\right) * 100$. The graph in Fig. (5) shows time

comparison of our PSD protocol with and without encryption for different sizes of our dataset. From the results, we can find that the proposed PSD protocol has a reasonable performance and the privacy preserving nature has marginal impact on the execution time in comparison with non encryption option.

In order to measure the correctness of our solution to capture correlated interests between attendees. We extracted sample data from conference proceeding related to 500 authors and co-authors. We crawled authors' website to create public profiles for them. Our aim here is to determine if our proposed solution can group attendees in the same sub-community and help them to find the right people to communicate or work with. For every sub-community recommendation for each participant in the conference, we need to test whether or not participants knew each other in this sub-community from previous work and if this recommendation accurate or not. Fig. (6) shows a breakdown of the results by our protocols, the percentage of unknown attendees recommended by *EMCP* are shown above the horizontal center line and the percentages of co-authors below. The chart also shows the percentage of accurate versus inaccurate in two different colors. PCF algorithm recommends other participants than the co-authors, which is not surprising because it mostly creates communities considering only similar interests without take in considerations the correlations between these preferences. In contrast, applying PCF and PSD extract sub-communities for people that are likely similar as sub-communities relies heavily on associations between preferences. These results confirm our intuitions that the more associations between participants' preferences, the more accurate sub-communities are produced.

In the last experiments, we evaluated the proposed solution from different aspects: privacy achieved and accuracy of results. We used precision and recall metrics proposed in [8] to measure privacy and accuracy of the results, The results are shown in Fig. (2). As we can see, a good quality is achieved due to: identifying communities that involve different sub-communities enables accurate recommendations to the attendees who share the same interests. Also, the effect of each interest inside the community can be easily measured, which enables to detect and remove outlier values that are very different than the general interests. We also evaluated the leaked private interests of different attendees when running our solution. We consider users, who published portion of their real interests in their public profiles, for each of these users; we tried the attack procedure proposed in threat model to reveal other hidden interests in their profiles based on the sub-community they belong. The obtained interests are quantified using our proposed metrics and the results are shown in Fig. (2). As we can see, our solution manages to reduce privacy leakages for exposed attendees' private interests, However, the revealed interests are only a hashed hypernym terms for attendees private interests.

7 Conclusion And Future Work

In this paper, we presented our attempt to develop an enhanced middleware for collaborative privacy for community based recommender service in conferences or exhibitions. We gave a brief overview of *EMCP* architecture and proposed protocols. We tested the performance of the proposed protocols on a real dataset. The experimental and analysis results show achieving privacy in recommending sub-communities is feasible under the proposed middleware without hampering the accuracy of the recommendations. A future research agenda will include utilizing game theory to better formulate user groups, sequential preferences release and its impact on privacy of whole profile.

ACKNOWLEDGMENT

This work partially supported by the European Comission via the ICT FP7 SOCIETIES Integrated Project (No. 257493). Also it was partially supported from the Higher Education Authority in Ireland under the PRTLI Cycle 4 Programme, in the FutureComm Project (Serving Society: Management of Future Communications Networks and Services).

References

[1] He, J., Chu, W.W., Liu, Z.: Inferring privacy information from social networks. Proceedings of the 4th IEEE international conference on Intelligence and Security Informatics. Springer-Verlag, San Diego, CA (2006) 154-165

[2] McSherry, F., Mironov, I.: Differentially private recommender systems: building privacy into the net. Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, Paris, France (2009) 627-636

[3] Esma, A.: Experimental Demonstration of a Hybrid Privacy-Preserving Recommender System. In: Gilles, B., Jose, M.F., Flavien Serge Mani, O., Zbigniew, R. (eds.), Vol. 0 (2008) 161-170

[4] Canny, J.: Collaborative filtering with privacy via factor analysis. Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, Tampere, Finland (2002) 238-245

[5] Canny, J.: Collaborative Filtering with Privacy. Proceedings of the 2002 IEEE Symposium on Security and Privacy. IEEE Computer Society (2002) 45

[6] Miller, B.N., Konstan, J.A., Riedl, J.: PocketLens: Toward a personal recommender system. ACM Trans. Inf. Syst. 22 (2004) 437-476

[7] Rubinstein, I.: Regulating Privacy by Design. Berkeley Technology Law Journal, Forthcoming (2011)

[8] Elmisery, A., Doolin, K., Botvich, D.: Privacy Aware Community based Recommender Service for Conferences Attendees. 16th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems. IOS Press, San Sebastian, Spain (2012)

[9] Elmisery, A., Botvich, D.: Privacy Aware Recommender Service using Multi-agent Middleware- an IPTV Network Scenario. Informatica 36 (2012)

[10] Elmisery, A., Botvich, D.: Enhanced Middleware for Collaborative Privacy in IPTV Recommender Services Journal of Convergence 2 (2011) 10

[11] Elmisery, A., Botvich, D.: Privacy Aware Recommender Service for IPTV Networks. 5th FTRA/IEEE International Conference on Multimedia and Ubiquitous Engineering. IEEE, Crete, Greece (2011)

[12] Elmisery, A., Botvich, D.: Multi-agent based middleware for protecting privacy in IPTV content recommender services. Multimed Tools Appl (2012) 1-27

[13] Elmisery, A., Botvich, D.: Privacy Aware Obfuscation Middleware for Mobile Jukebox Recommender Services. The 11th IFIP Conference on e-Business, e-Service, e-Society. IFIP, Kaunas, Lithuania (2011)

[14] Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. 34 (2002) 1-47

[15] Yao, A.C.: Protocols for secure computations. Proceedings of the 23rd Annual Symposium on Foundations of Computer Science. IEEE Computer Society (1982) 160-164

[16] Beil, F., Ester, M., Xu, X.: Frequent term-based text clustering. Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, Edmonton, Alberta, Canada (2002) 436-442

[17] Fung B. C, M.: Hierarchical document clustering using frequent item sets. Master's Thesis, Simon Fraser University (2002)

[18] Cheung, D.W., Han, J., Ng, V.T., Fu, A.W., Fu, Y.: A fast distributed algorithm for mining association rules. Proceedings of the fourth international conference on on Parallel and distributed information systems. IEEE Computer Society, Miami Beach, Florida, United States (1996) 31-43

[19] Cuesta-Frau, D., Pérez-Cortés, J.C., Andreu-García, G.: Clustering of electrocardiograph signals in computer-aided Holter analysis. Computer methods and programs in biomedicine 72 (2003) 179-196

Appendix D: Community Discovery & Recommendation Service Scenario

Article XIII

Privacy Enhanced Middleware for Location based Sub-Community Discovery in Implicit Social. Groups

Ahmed M. Elmisery, S. Rho, Dmitri Botvich

Accepted for The Publication in the Springer International Journal of Electronic Commerce Research, September 2014.

Privacy Enhanced Middleware for Location based Sub-Community Discovery in Implicit Social Groups

Ahmed M. Elmisery[1,*], S. Rho[2] and Dmitri Botvich[1]

[1]TSSG, Waterford Institute of Technology-WIT-Co. Waterford, Ireland

[2] Department of Computer Science, Kyungpook National University, Taegu, South Korea

Email: ahmedmohmed2001@gmail.com

* Corresponding Author

Abstract

In our connected world, recommender services have become widely known for their ability to provide expert and personalize referrals to end-users in different domains. The rapid growth of social networks, a new kind of services so called "community based recommender service" are raising, where recommender services can be utilized to discover sub-communities from users groups and provide referrals for new users to join various sub-communities of other users with similar preferences. However, preserving end-users privacy in recommender services is a very challenging problem that might prevent end-users from releasing their own data which detains the accuracy of generated referrals. So in order to gain accurate referrals, recommender services should have the ability to discover previously unknown users sub-communities from different social groups in a way to preserve privacy of end-users. In this paper, we present a middleware that runs on end-users' mobile phones to sanitize their profiles data when released for generating referrals, such that computation of recommendation proceeds over the sanitized data. The proposed middleware is equipped with cryptography protocols to facilitate discovering sub-communities of participants within university scenario in private way. Location data is added to users' profiles in order to improve the awareness of surrounding sub-communities so the provided referrals can be based on current location of a participant. We performed a number of different experiments to test the efficiency and accuracy of our protocols. We also developed a formal model for the tradeoff between privacy level and accuracy of referrals. As supported by the experiments, the sub-communities were correctly identified with good accuracy and acceptable privacy level.

Keywords: Privacy; Clustering; Community Recommendations; Middleware

1 Introduction

The dynamics and evolution of social groups have been studied by different researchers. One of the most important approaches in this field is community discovery process. Whereas, communities within social groups can be extracted based on certain properties such as shared interests, shared purposes, shared behaviors and shared space-time continuum. The concept of community can be defined in different ways; at a high level we can consider it as set of "similar people". Similarity will determine the type of community we are discovering. Therefore, our aim is to detect a set of people who are similar to each other on some pre-defined type and less similar to people in other sets. Communities can be divided into either explicit or implicit, where explicit communities can be easily detected since they have a formal structure such as faculty members or staff. On the other hand, implicit communities are harder to detect due to their volatile and semi-permanent nature such as study group or car sharing group. These communities are formed based on mutual interests between participants that might be unknown to other communities. Virtual communities have the same properties like discussed above, but they are built using web technologies while maintaining privacy preferences of their participants. In web 2.0 era, service providers employ community aware mechanisms on the personal information about consumers that are gathered through various means [1], in order to be able to personalize their services and offers to specific communities. Moreover, having communities of end-users provides better understanding of social behavior while utilizing service and helps to identify interaction patterns between users of different communities.

The importance of recommender service is increased due to the information overload [2] that causes amplitude of choices when making decisions related to any human activity. Recommender service can be used to provide precious and expert advice to lessen the list of available options in short time; in order to aid users in electing suitable choices while the final decision will be left to the end-user. These services have been successfully employed in different contexts starting from recommending books or products, up to recommending persons to socialize with or date. However, with the advancements in mo-

bile, wireless and positioning technologies [3] employing location awareness with recommender service empowers referrals with location awareness feature, which gives opportunity to create a location aware sub-community recommender service that permits the usage of participants' locations to see the locality of nearby sub-communities within certain plane. The popularity of these services is increasing rapidly with the spread of cheap handed devices equipped with GPS sensors. However, these services are always eager to collect tremendous amount of precise data from different participants' profiles, which raises privacy concerns for participants that might prevent them from releasing their data. Especially when implementing location awareness feature in a recommender service, it is required that participants agree to reveal their real locations, which poses a severe threat to their privacy.

From privacy point of view, many of the current recommender services have failed to meet the privacy requirements of end-users, which result in a lack of acceptance of the respective services in general as the more information is revealed to the recommender service the lower privacy levels can be guaranteed. The raise of Privacy as an essential concern is a direct result of unauthorized procurement and exploitation of personal information from the use of these services which in turn discourages users from disclosing their information or encourages users to submit fake information online [4] that in turn reduce the referrals' accuracy. Therefore, giving precedence to end-users privacy bigger benefits can be reaped by all parties involved [5].

Several strategies have been proposed to control the disclosure of private information. The most popular approach is to permit users to maintain a set of privacy rules, according to which a decision is performed whether a certain user(s) is/are able to view certain preferences in owner profile. However, these approaches are often not sufficient enough to protect users' privacy as the least secure of these approaches are the weakest link in the chain and the threat to a person's information is only as strong as this weakest link. Since it is not a feasible task to ensure that all services that store personal information are at a certain level of security, furthermore users are the weakest link in any security chain [6] . These approaches are either rather coarse-grained, or require a deep understanding of the privacy control system, any change of one privacy setting may result in unwanted or unexpected behaviors. Although in some occasions it is possible to gain a relatively fine-

grained privacy policy, it requires through understanding of the whole privacy control system and quite often it may take a huge amount of time to make a good privacy policy which is undesirable in the case where users have limited resources to process privacy settings. Moreover, these approaches are based on the logic of either to allow or deny releasing certain attributes in users' data. Once, the data is released the user have no control over it and users will be vulnerable for the privacy breaches since released pieces of users' information is often interleaved, adversaries may be able to infer other private information using inference techniques. For example work in [7] shows that private information can be inferred via social relations, and the stronger the relationships people have in the network, the higher inference accuracy can be achieved. The main concern of this paper is to limit the secondary use of users' data and what is the level of control the user has over it is the main concern of this paper. Even though many nations have developed privacy protection laws and regulations to guard against private use of personal information, the existing laws and their conceptual foundations have become outdated because of changes in technology [8].

In this paper, we propose a scenario where community based recommender service (CRS) is employed to discover sub-communities between university students based on their profiles, and then it offers referrals for new students to join different sub-communities. Neighboring participants' profiles will be used to form initial sub-communities until no changes occur to them. Then merging process is employed on all profiles belonging to the same initial sub-community in order to obtain a final sub-community. Once sub-communities are detected, CRS possesses certain information about these sub-communities from previous runs. CRS can offer referrals to new participants, considering both their location and preferences data. Hence the produced referrals are composed of a list of interesting sub-communities along with their distances from current participant's location. This is a realistic requirement for our service, which facilitates the identification of sub-communities and their meeting places. We note that, finding a specific sub-community in large plane is a challenging task to achieve in short time. The distances between these sub-communities and participants are rendered in the form of two states either "close" with distance or "separated". Such that, when a participant is in "close" state, he/she wants to know the approximated distance to a specific sub-

community, but if he/she in "separate" state, a participant knows that he/she is away from any sub-community. The locations of sub-communities are filtered in such away to reveal only the ones that have profiles similar to participant's profile. This will help to hide the locations of other mismatched sub-communities.

In order to preserve the privacy of participants in CRS, we proposed an enhanced middleware for collaborative privacy *(EMCP)* that allows participants to release their data for sub-communities discovery and recommendation processes without breaching their privacy. *EMCP* employs a set of cryptography protocols to allow participants to share their data among each other in the network and obtain referrals for joining various sub-communities in a private way. The first protocol is the private community formation (PCF) which builds general communities between participants, such that each general community contains various participants who share the same preferences. The second protocol is private sub-community discovery (PSD) that helps to discover sub-communities inside each community that is extracted using the PCF. Secure distance detection (SDD) is the third protocol which runs in distributed manor in order to permit participants to detect nearby permanent or newly formed sub-communities and their meeting places while respecting their location privacy.

The participants' cooperation is needed not only to protect their privacy but also to allow the recommender service to run properly. Highly reputable peers are selected to aggregate and perform computations on encrypted participants' profiles. These profiles are encrypted using homomorphic encryption in order to permit particular operations to be performed on them without need for prior decryption. Moreover, participants cannot trust each other as well and hence the aggregation process should not expose their preferences data. As a result, *EMCP* permits encrypting and aggregating profiles to hide the identities of participants, and thus hamper the ability for the untrusted parties to invade participants' privacy by profiling or tracking them. Hence, participants can create hashed public (generalized) profile from their private profiles in order to facilitate creating general communities of shared interest where they can share & aggregate their encrypted private profiles before publishing them to the highly reputable peers (super-peers). This paper is organized as follows. In Section 2, related works are described. Section 3 presents the threat model used in this work .Section 4 introduces community based recommender ser-

vice (CRS) that is landing *EMCP*. An overview of cryptography tools used in our protocols is presented in Section 5. The proposed protocols that are used in *EMCP* are introduced in detail in Section 6. Tradeoff between privacy and accuracy of generated referrals of our solution is discussed in Section 7. In Section 8, the results from some experiments on the proposed mechanisms are presented. Finally, the conclusions and future work.

2 Related Works

Building users profiles in recommender services can be performed either based on ratings (explicit rating procedures) or on log archives (implicit rating procedures) [9]. The majority of the literature addresses the problem of privacy on social recommender service, as it is a potential source of leakage of private information shared by the users as shown in [10]. In [11] a descriptive framework for personalization aspects of e-businesses, in business-to-consumer (B2C) situations, that is related to typical e-business functionality. This framework classifies previous research and extends it to provide e-commerce stakeholders with a vocabulary for analyzing e-businesses, for comparing personalization features, and for explaining e-business commerce evaluation results. In [12] a theoretical framework is proposed to preserve the privacy of customers and the commercial interests of merchants. Their system is a hybrid recommender system that uses secure two party protocols and public key infrastructure to achieve the desired goals. In [13] a protocol to protect data collected by mobile agents roaming through a set of potentially malicious hosts. This protocol is based on an original secure cryptographic technique that assures the integrity of a sequence of data segments regardless of the order of each segment in the sequence. In [14, 15] a privacy preserving approach is proposed that is based on peer to peer techniques using users' communities, where each community has an aggregate user profile representing the group as a whole but not individual users. Personal information is encrypted and communication done between individual users but not servers. Thus, the recommendations are generated on the client side. In [16, 17] another method is suggested for privacy preserving in centralized recommender systems that adds uncertainty to the data using a randomized perturbation technique while attempting to make sure that the necessary statistical aggregates such as the mean do not greatly get

disturbed. Hence, the server has no knowledge about the true values of the individual items' ratings for each user. They demonstrate that this method does not essentially decrease the accuracy obtained in the results. But recent research work [18, 19] pointed out that such techniques do not provide levels of privacy as was previously thought. In [19] it is pointed out that arbitrary randomization is not safe because it is easy to breach the privacy protection it offers. They proposed random matrix based spectral filtering techniques to recover the original data from the perturbed data. Their experiments revealed that in many cases, random perturbation techniques preserve very little privacy. Storing users' profiles on their own side and running the recommender system in a distributed manner without relying on any server is another approach proposed in [20], where the authors proposed only transmitting similarity measures over the network and keeping users' profiles secret on their side to preserve privacy. Although this method eliminates the main source of threat against user's privacy, it requires higher cooperation among the users to generate useful recommendations.

3 Threat Model

The proposed solution is secure in an honest-but-curious model. We assume that an adversary aims to collect participants' preferences and location data in order to identify and track students. Thus, we consider our main adversary to be an untrusted CRS; moreover we do not assume CRS to be completely malicious. This is a realistic assumption because CRS needs to accomplish some business goals and increase its revenues. CRS can construct the profiles of the students based on a leak of shared preferences between sub-communities members. Moreover, CRS can infer student's location, once he/she runs the location awareness feature. Hence, our aim is to detain the ability of the adversary to identify students' private preferences and locations based on a set of shared preferences between sub-communities members and released locations.

4 Proposed Community based Recommender Service in University Scenario.

In this section, we present our scenario and analyze the issues related to the privacy of students' preferences in community building process. Close inter-student interactions have raised a new concern on the privacy of their preferences; which is the key challenge

in community building process due to the diversity and massive size of students-generated profiles. The scenario we are targeting here can be summarized as follows, Staff in student affairs department can employ a community based recommender service in order to facilitate social and educational interaction between various students from different backgrounds. For example, students can use this service to form a study group for specific subjects, a group to have lunch together, a team to play football together or residents in neighboring areas can form a group to make shopping together. These groups (sub-communities) tend to evolve out of collaborating members with similar preferences and the participation of new members for example new international students can discover sub-communities of similar interests on their campus and join them. Moreover, this service can aid faculty members to detect sub-communities of diligent students who have regular research discussions after lectures and have scientific curiosity, facilitates inferring the association between sub-communities and their locations for example this information can aid in discovering places that need improvement on campus by using statistical information about meeting locations and offering customized notifications for different sub-communities based on their preferences for example basketball sub-community can be notified about up-coming training schedule during the exams. All these recommendations can be obtained from CRS while respecting privacy constraints and requirements of participants by enabling them to control over what parts of their profiles they are willing to share and in which granularity.

The architecture of our solution (CRS) is depicted in Fig. (1). CRS provides students with personalized referrals for joining specific sub–communities that are similar to his/her profile. In this architecture, a participant's profile is used to capture the updated preferences of different students in the campus. It typically includes demographics information such as city of birth, address of residence, courses profile, etc besides other interests or preferences for student himself/herself for joining various sub–communities. Each extracted community consists of various sub-communities profiles where each profile keeps track about all data related to each sub-community. This data represents the collective preferences for different members of this sub-community besides topic of discussions. Moreover, each sub–community profile has a location representative that illustrates current location for this sub-community and it can be used in guiding a new participant to

location of this sub-community within a specified distance from his/her current location. Every sub–community size can vary significantly as it contains a group of students who are likely to be close to each other. For instance, a study sub-community could consist of 4-6 members, while a football sub-community could consist of 22-44 members. Additionally, members could join multiple sub-communities and they can join and leave sub-communities frequently.

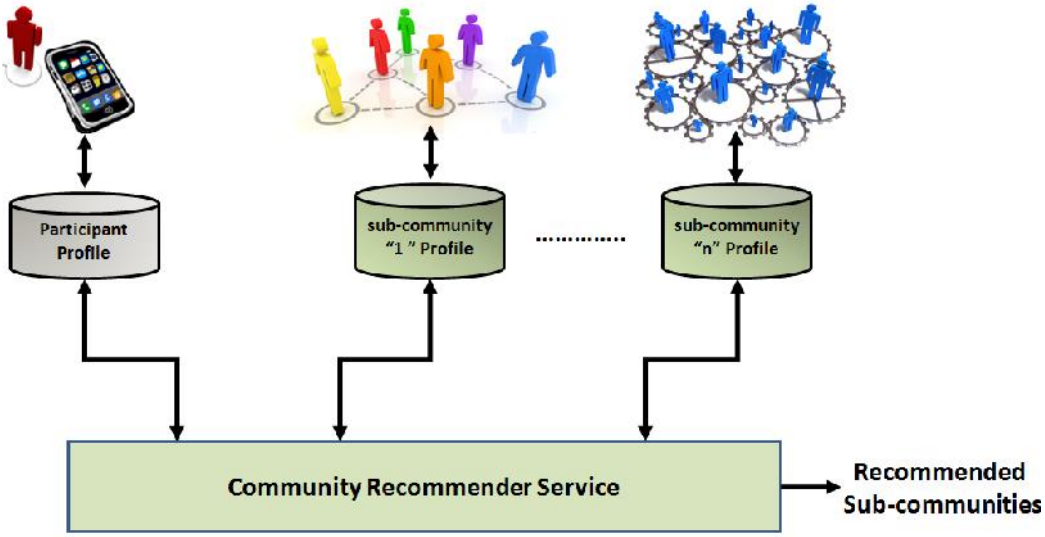


Fig. 1. Generating Recommendations for Participants

Existing students employ *EMCP* to release their profiles in an obfuscated form to super-peers; then these super-peers collaborate in categorizing different communities that contain various sub-communities profiles. Moreover, these profiles assist CRS to offer referrals to new students based on the similarity between their profiles and these sub-communities profiles and the distances between these sub-communities from participants' current location. Assigning a new student to sub-community could implicitly update the formulated sub-communities profiles. In the case, a new student profile does not have enough preferences for generating referrals; recommendations can be made using his/her demographics information after anonymizing it.

4.1 Interaction Sequence Between Participants and CRS

Based on various topics and activities in the university, faculty members propose different communities each of which has its own interaction space where any interactions are supported. Each student configures his/her *EMCP* to build a public profile that discloses some generalized information about their preferences for socializing and collaboration, Moreover, students can contribute to the proposed topics and activities. Students seek to hide from the public their personal information, sexual preferences, specific expertise, previous study history, details of their study/research domains and problems in hand, their time tables, and finally their arrival/departure times. Private information such as names, addresses, etc, by default is protected by the privacy protection rules. In some cases where students already belong to previously created sub-community, they can form the same sub-community inside the campus, such that they can participate in discussions while having access to the previously exchanged opinions. CRS provides information that can help faculty members and staff as explained before. Moreover, it provides students with personalized location based referrals to lead them to their desired sub-communities. The interaction sequence for sub-community discovery looks as follows:

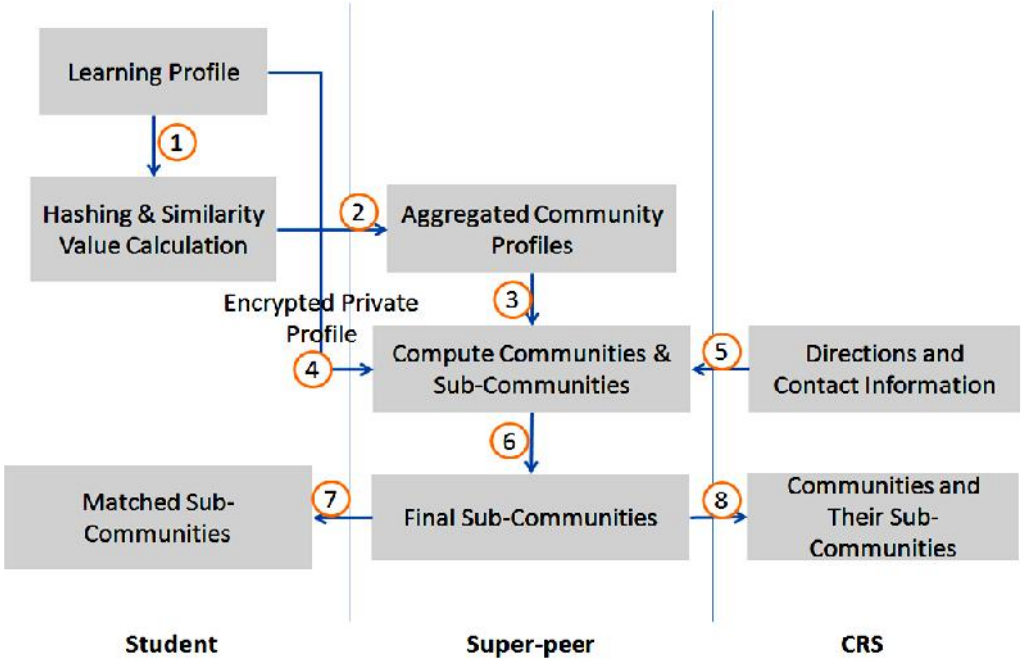


Fig. 2. Interaction Sequence Diagram of CRS

1. Staff in student affairs department setup a set of general communities that represent the general themes in the university campus. Then they submit a request to CRS to start the process of forming communities from students group.
2. CRS accepts the request after authenticating the staff's credentials, and then it selects peers with highest reputations [21] from each students group which will serve as super-peers for collecting profiles from students group. Then, CRS submits the request to these super-peers.
3. The super-peers coordinate with their members to aggregate their hashed public profiles in order to create different communities using private community formation (PCF) protocol.
4. After extracting communities, participants of each community encrypt their hashed private profiles then engage in peer to peer communication between other members to execute private sub-community discovery (PSD) protocol to discover the best sub-community that matches their expertise. Then they submit results to super-peers in order to form suitable sub-communities. Super-peers submit discovered sub-communities representatives to both CRS and community members.
5. For a new student who wants to join a sub-community, *EMCP* contacts CRS and obtains a list of sub-communities representatives and their locations. Then it encrypts its host private profile and then it engages in matching protocol with these representatives. This process will yield to one encrypted value for each sub-community that indicates the degree of similarity between each sub-community profile and participant's profile. In order to privately recommend suitable sub-communities for new student, *EMCP* selects top N sub-communities with the highest similarity values and offers this list to the student to select suitable ones. Thus we manage to achieve both high recommendation quality and good privacy preservation.

4.2 Proposed Middleware for collaborative privacy

Our presented middleware [22-28] presents a solution that mitigates the tension between privacy and accuracy of referrals. Namely, it enables CRS's functionalities while protecting sensitive profiles data.

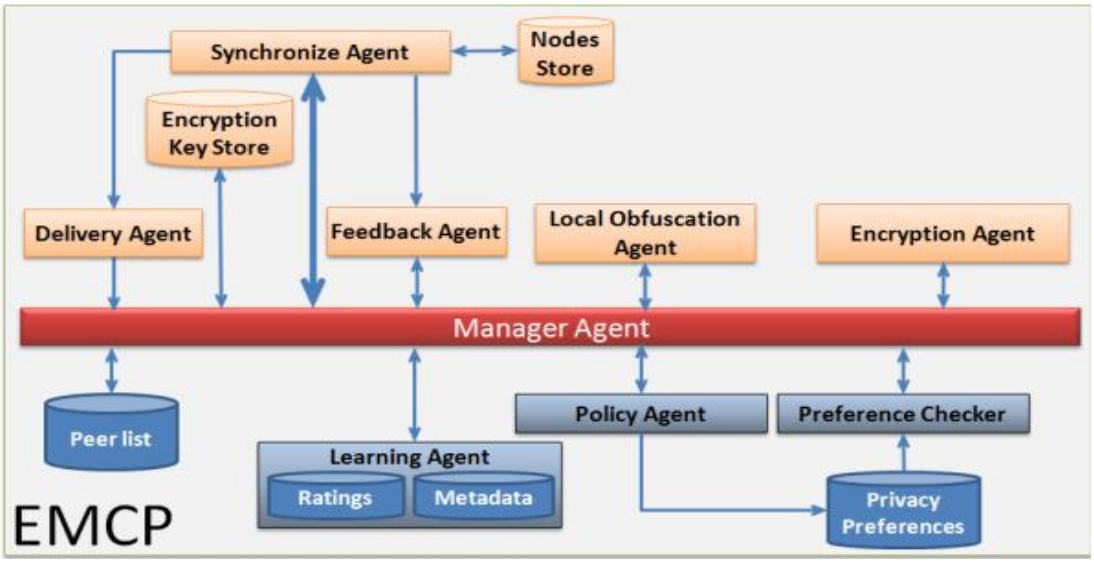


Fig. 3. *EMCP* Components

EMCP is implemented as a middleware running on top of students' mobile phones, a more precisely architecture of this middleware is presented in Fig (3). *EMCP* consists of different co-operating agents, a learning agent captures students' preferences explicitly or implicitly to build two databases one for preferences and other for meta-data. The local obfuscation agent creates a generalized profile that is used as an input to encryption agent. The encryption agent is responsible for executing three cryptographic protocols; first one is the private community formation (PCF) protocol which builds communities between students, while the second one is the private sub-community discovery (PSD) protocol that helps to discover sub-communities inside each community. The last one is secure distance detection (SDD) protocol which runs in distributed manor in order to permit participants to detect nearby sub-communities and their meeting places while respecting their privacy. These protocols act as wrappers that conceal profiles' data before they are shared with any external entity. Since the database is dynamic in nature, the local obfuscation agent periodically creates generalized entries for the updates in profile, and

then a synchronize agent forwards them to CRS upon owner permissions. Thus communities will be always updated. The policy agent is an entity in *EMCP* that acquires the participant's privacy preferences and expresses them using APPEL as a set of preferences rules which are then decomposed into a set of elements that are stored in a database called "privacy preferences" as tables called "privacy meta-data". These rules contain both a privacy policy and an action to be taken for such privacy policy. In such a way this will enable the preference checker to make self-acting decisions on objects that are encountered during data collection process. For example, privacy preferences may include: certain preferences that should be excluded from data before submission, generalizing certain preferences in according to a predefined taxonomy, using synonyms for certain preferences and inserting dummy preferences that have same feature vector like the suppressed ones. *EMCP* requires students to be organized into virtual topology which may be a simple ring topology or hierarchical topology. This ordering enables them to participate in multi-party computations as well. However, CRS is the server that initiates the process to extract different communities and sub-communities. So, any new student who wants to find suitable sub-communities submits a request to CRS.

5 Cryptography Tool

1. **Additively homomorphic cryptosystem** permits the computation of linear combinations of encrypted data without need for prior decryption. Formally, an encryption schema $E(.)$ denotes the encryption function with encryption key pk and $D(.)$ denotes the decryption function with decryption key sk. Additive homomorphic cryptosystem possesses the following properties:
    - Given the encryption of plaintexts m_1 and m_2, $E(m_1)$ and $E(m_2)$. The sum $m_1 + m_2$ can be directly computed as $E(m_1 + m_2) = E(m_1) * E(m_2)$.
    - Given a constant k and the encryption of plaintexts m_1, $E(m_1)$. The multiplication of k with the plaintext m_1 can be directly computed as $E(k.m_1) = E(m_1)^k$.

    Paillier [29] proposed a probabilistic asymmetric algorithm for public key cryptography that is an example of an efficient additively homomorphic cryptosystem.

2. **Cryptographic hash function** is an efficient one-way function that does not require the use of a secret key and is preimage resistant: given h, it is difficult to find any M such that $\mathrm{hash}(\mathrm{M}) = \mathrm{h}$. We will employ hash function to hide public interests while exchange profiles data. To be more precise, an attendee $\mathrm{v_b}$, which wants to compare his/her public profile with $\mathrm{v_s}$'s profile, and then he/she would simply hash all elements of his/her public profile, thus obtaining: $H(D_{v_s}) = \left\| \forall_{j \in L} \left(ID_j, hash\left(A_j^{v_s}\right) \right) \right\|$. As a counterpart, an intermediate participant $\mathrm{v_c}$ could compare his/her interests with $\mathrm{v_s}$'s interests in two steps, first $\mathrm{v_c}$ would first hash its profile with the same hash function hash. Then $\mathrm{v_c}$ tests whether one of its hashed interests $\mathrm{hash}\left(\mathrm{A_j^{v_c}}\right)$ is equal to an interest $\mathrm{hash}\left(\mathrm{A_j^{v_s}}\right)$ of the sender $\mathrm{v_s}$. With help of the preimage resistance of hash, this implies that the interests where the equality holds are shared interests and the one where it does not hold are non-matching interests. Hash functions are efficient to compute thus it can be a public function in our middleware. Moreover, this idea is effective as the further computations of our protocols require only the size of interaction between various attendees.

6 Problem Formulation

In the following section we explain notions used in our solution, attendees' profiles can be represented in two categories public profiles $P(v)_{pub}$ and private profile $P(v)_{priv}$. Our goal is to protect their private interests when formulating communities and recommending sub-communities to them. Specifically, we focus on protecting private participants' profiles since these are the information that attendees do not disclose to public and wish to keep private against both PCRS and any third parties.

Definition 1. Our model is defined as follows: $v, i, level$ represents (participants, interests, access level) respectively. While permission $i \times level \rightarrow \{hypernym, Restrictive\}$ is a function that answers if an interest i can be published by *EMCP* in order to help him/her finding a suitable community.

Definition 2. (Synonym set) A synonym set of an interest is a set of words and phrases including the interest and all its synonyms. Synonym set introduces different lexical

forms for attendees' interests. For example, {data mining, predictive analytics, statistical analysis} is the synonym set for any interest within this set.

Definition 3. (Hypernym path) In linguistics, a hypernym is a word or phrase whose semantic range includes that of another word, its hyponym. Hypernym path for a synonym set is a list of synonym sets including the root synonym set and all its hypernym sets. For example, data mining, predictive analytics, statistical analysis, are all hyponyms of machine learning (their hypernym).

Definition 4. The public profile of user v is a set of hypernym terms in the same semantic categories for the interests in attendee's profil. $P(v)_{pub} = \{i_i, i_2, \dots, i_n\} \forall\, i \in P(v)_{pub} \equiv$ user v assigns $i.level = hyperny$. The similarity between the terms on the public profile and the attendees' profile is computed then select the one that has the highest similarity.

Definition 5. The private profile of user v is a set of ests $P(v)_{priv} = \{i_i, i_2, \dots, i_m\} \forall\, i \in P(v)_{priv} \equiv\ i \notin P(v)_{pub}$ && $user\ v$ assigns $i.level =$ Restrictive||$\emptyset$.

In other words, we define $P(v)_{pub}$ as the generalized information that a user v configures his/her *EMCP* to disclose, while $P(v)_{priv}$ represents the "hidden" information that v does not want to disclose publically to others.

Definition 6. A community is the set $C = \{c_1, c_2, \dots, c_n\}$, where n is the number of sub-communities in C, has the following properties: (1) Each $\forall_{i=1}^{n} c_i \in C$ is a 3-tuple $c = \{I_c, V_c, d_c\}$ such that $I_c = \{i_1, i_2, \dots, i_l\}$ is a set of interests, $V_c = \{v_1, v_2, \dots, v_k\}$ is a corresponding set of attendees, and $d_c \in I_c$ is the centroid of c. (2) For each attendee $\forall_{i=1}^{l} v_i \in V_c$, v have the interests V_c. (3) Centroid d_c has the smallest average distance from other preferences in V_c, and it represents the "core-point" of sub-community c. (4) For any two sub-communities c_a and c_b $(1 \leq a, b \leq n$ and $a \neq b)$, $V_{c_a} \cap V_{c_b} = \emptyset$ and $I_{c_a} \neq I_{c_b}$.

7 Proposed Protocols

EMCP is equipped with three cryptography protocols to facilitate sharing of data between various participants and performing computation on encrypted data during communities

and sub-communities building and recommendation processes. In the followings sub-sections, we give overview on these protocols.

7.1 Private community formation (PCF)

EMCP utilizes PCF protocol proposed in [28] to cluster students into general communities, such that each general community contains various participants who share the same preferences. A student can belong to multiple communities with various public profile based on his/her private profile. PCF ensures participants privacy when forming communities and matching their public profiles with the list of available communities. PCF can be summarized as follows:

- For any attendee $v \in V$ and a set of interests I, v denote possession of an interest $i \in I$ as $P_{v,i} = 1$ and 0 otherwise. Dice similarity is calculated in two steps first, it computes the numerator $\left|V_{I_A} \cap V_{I_B}\right|$ a between each attendee v_A and v_B and then it computes the denominator$\left|V_{I_A}\right|^2 + \left|V_{I_B}\right|^2$.
- After selecting a super-peer, a ring topology is employed for calculating the numerator between every two attendees' public profile. Attendees who are willing to participate hash their interests $\{I_j\}_{1\le j\le m}$ where each interest I_j defined with name N_j and value U_j. The attributes set $\{N_j\}_{1\le j\le m}$ are known, while set $\{V_i\}_{1\le i\le p}$ have different values. After subset extraction, B compute $\left|V_{I_B}\right|^2$ then it encrypt these values along with participants' pseudonyms identities using super-peer public key and forwards them with A's profile to the next participant in his group.
- Super-peer collects all these results and decrypts them with its private key. Then it starts to cluster participants into communities. A participant can belong to multiple communities based on his/her public profile. Super-peer performs S-seeds clustering algorithm to identify the communities based on distance calculations between hashed public profiles.

7.2 Private sub-community discovery (PSD)

EMCP utilizes PSD protocol proposed in [28] to extract sub-communities from the proximate general communities extracted from the first process. PSD is executed in a bilateral

manor to match the preferences within student's private profiles. Then the final results are used in building sub-communities. PSD can be summarized as follows:

- Each attendee A and B apply a hashing function lsh to encode their private profiles data and generate $E_A = lsh(D_A)$ and $E_B = lsh(D_B)$. In the same time, each attendee in sub-community engage in distributed threshold key generation process with other attendees to generate a complete public key PK along with a share of the private key SK.
- Attendee A encrypts independently his hashed data with PK $(E_A)_{PK}$ and sends it to other attendees in his community. For simplicity we assume that A sends his data to attendee B. Attendee B compute s $(R_{AB})_{PK} = \prod_{i=1}^{n} \left(\left(E_i^A\right)_{PK} (E_i^B)_{PK} \left(\left(E_i^A\right)_{PK} \right)^{-E_i^B} \right)$. Furthermore, B select random value t and multiply it in the encryption $(tR_{AB})_{PK} = ((R_{AB})_{PK})^t$ then send the result to its next neighbor till the last attendee who submits the final result to the super-peer in the community.
- After super-peers receive encrypted values, they start calculating the final similarity values for the members in the entire community without decrypting these values, moreover decryption process requires number of participants to cooperates, Super-peer then computes $|A \cap B|$ based on counting the number values $== (0)_{PK}$ then it calculate the similarity between sets A and B using Jaccard similarity $|A \cap B|/|A \cup B|$, a matching declared if similarity value $\geq S$. Super-peers formulate a list (ε, S) of sub-communities such that each sub-community has a center ε which represents the centroid vector for it. Super-peers publish this list to CRS and participants too

Proof of security for PCF&PSD Protocols.

in our work, we assume honest but curious model, in which attendees follow the rules of our protocols properly without any deviations and provide the correct inputs with the exception that they might keep a record of all intermediate values of computations. This model is a realistic assumption, as adversaries cannot obtain further information about different attendees, since public profiles are a hash generalized version of original profiles and private profile are hashed and encrypted with threshold key that requires cooperation between all community members (including victim him/herself) to obtain the real

hashed profiles. None of the participants have decryption key; sub-communities themselves are represented using centroid and similarity value, Moreover attendees' public/private profiles are omitted from submission to PCRS. In the honest but curious model, the achieved privacy can be proven formally, but due to the page limits, we omitted it.

7.3 Secure Distance Detection (SDD)

Secure distance detection (SDD) protocol combines both cloaking strategy with secure multiparty computation in order to be able to permit participants to directly detect surrounding sub-communities without revealing their real location and selects the matching ones to their profiles with minimum location update. Cloaking strategy proposed in [30] is employed where each participant setups a circle with specific radius around his/her current location. This cloak hides user's real location by unitizing any imprecise location within that circle for any further computation requires location data. Therefore the only information that is revealed is this imprecise location. Secure multiparty computation is used to calculate the distance between participant's location and similar sub-communities. SDD is inspired from work proposed in [31, 32] to computes the distance between participant and elected representatives without revealing their real locations. Each sub-community representative setups a circle (cloak) with radius R_{C_i} around its current location l_{c_i} with guarantees that the distance between any of its m_i members' locations $\forall_{i=1}^{m} l_{p_i}$ and l_{c_i} falls below distance D_{C_i}, $dist(\ l_{c_i}, \forall_{i=1}^{m} l_{p_i}\ ) \leq D_{C_i}$ where $D_{C_i} \leq R_{C_i}$. While a sub-community member is moving within this circle, no updates are done and state stays "close", in case if any member makes a movement bigger than D_{C_i}, he/she notifies the sub-community representative about his/her about this update. SDD Protocol can be summarized as follows:

1. After participant P_i gets referrals list from CRS, he/she configures *EMCP* to setup a circle with radius R_{P_i} around his/her current location l_{P_i} and generates a key pair (E_{P_i}, D_{P_i}). Then, *EMCP* selects any imprecise location $\acute{l}_{P_i}$ from this circle to represent current location then it computes $E_{P_i}(\ \acute{l}_{P_i}) = E_{P_i}(x_{P_i}), E_{P_i}(y_{P_i})$, and sends $(E_{P_i}(\ \acute{l}_{P_i}), E_{P_i})$ to each representative.
2. *EMCP* at each representative side randomly generates an imprecise location from its predetermined circle $\acute{l}_{c_i} = (x_{c_i}, y_{c_i}) \in Circle(R_{C_i})$, and also two random num-

bers o, o_1. Then it computes $E_{P_i}(\acute{l}_{P_i})^{\acute{l}_{c_i}} = E_{P_i}(x_{P_i})^{x_{c_i}}.E_{P_i}(y_{P_i})^{y_{c_i}}$ and $b = x_{c_i}^2 + y_{c_i}^2 + o + 2o_1$. Finally, it sends (a, b) to participant P_i where $a = E_{P_i}(\acute{l}_{P_i})^{\acute{l}_{c_i}}.E_{P_i}(o_1)$.

3. When *EMCP* at participant side receives (a, b), it decrypts a to obtain $D(a) = x_{P_i}.x_{c_i} + y_{P_i}.y_{c_i} + o_1$ and it generates random number z. Next, it computes $h = x_{P_i}^2 + y_{P_i}^2 - 2x_{P_i}.x_{c_i} - 2y_{P_i}.y_{c_i} - 2\,o_1 + b + z$ and sends the value h to sub-community representative in order to computes $g = h - o$ and sends this value g to participant back
4. *EMCP* at participant computes the distance between the released location and sub-community location which is equal to $\text{dist}(\acute{l}_{P_i}, \acute{l}_{c_i}) = \sqrt{g - z}$. If $\text{dist}(\acute{l}_{P_i}, \acute{l}_{c_i}) \leq D_{C_i}$ participant state is altered to "close" otherwise it is stay "separate".

When a new participant obtains a set of related sub-communities as referrals, he/she want to know their pertinent distances from his/her current location. So, *EMCP* runs SDD protocol in order to calculate distance between them. SDD protocol can determine distance between the released locations of different parties in a way to preserve the privacy of their locations. Privacy level increases when D_{C_i} value is higher, since minimum number of location updates is required from members and less accurate locations about them are saved. In the same time, cloaking radius determines the range of circle around each party location, so the increase of its radius attains high privacy but reduces the accuracy of distance computations since the uncertainty in these calculations are related to randomness of released locations. In the context of sub-communities discovery in CRS, we have encountered two cases that should be taken into consideration. The first case, if any member traverses circle borders of its sub-community and updates his/her location, his/her state is altered to separate then he/she had to setup a circle with radius UR_{P_i} around his/her current location l_{P_i}.The following algorithm refers to this update operation based on SDD protocol.

i. If participant P_i movement $> D_{C_i}$ then his/her *EMCP* runs SDD protocol to late $\text{dist}(\acute{l}_{P_i}, \acute{l}_{c_i})$ with the representative of sub-community he/she belongs.
ii. If $\text{dist}(\acute{l}_{P_i}, \acute{l}_{c_i}) > 2R_{C_i}$, P_i state is altered to "separate" and then *EMCP* computes a new radius UR_{P_i} for his/her new cloak, $UR_{P_i} = \frac{\left(\text{dist}(\acute{l}_{P_i}, \acute{l}_{c_i}) + R_{P_i}\right) - D_{C_i}}{2}$

In the second case, the sub-community representative along with plenty of its group members traverses their circle borders. The following algorithm refers to this update operation based on SDD protocol,

i. If the sub-community representative and its members movement $> 2R_{C_i}$&& number of moving members $|P_i|_{C_i} > \delta$ (where δ is defined number), *EMCP* at representative side runs SDD protocol to compute $dist(\ l_{c_i}, \forall_{i=1}^{m} l_{p_i}\ )$,

ii. A new cloak is formed for this sub-community with the updated radius $UR_{C_i} = \frac{max\left(dist(\ l_{c_i}, \forall_{i=1}^{m} l_{p_i}\ )\right) - D_{C_i}}{2}$.

iii. If only the sub-community representative traverses circle borders, a new representative is calculated between members to hold this role.

Proof of Security for SDD Protocol.

Theorem 1: Participant P_i knows relative distance between his/her location and each sub-community.

Proof: Step 2 calculates intermediate values required for calculating distance as follows:

$$E_{P_i}(a) = E_{P_i}(\acute{l}_{P_i})^{\acute{l}_{c_i}}.E_{P_i}(o_1) = E_{P_i}(x_{P_i})^{x_{c_i}}.E_{P_i}(y_{P_i})^{y_{c_i}}.E_{P_i}(o_1)$$

$$b = x_{c_i}^2 + y_{c_i}^2 + o + 2o_1$$

Then at Step 3 aggregates of these intermediate values are computed:

$$D(a) = x_{P_i}.x_{c_i} + y_{P_i}.y_{c_i} + o_1$$

$$h = x_{P_i}^2 + y_{P_i}^2 - 2x_{P_i}.x_{c_i} - 2y_{P_i}.y_{c_i} - 2\,o_1 + x_{c_i}^2 + y_{c_i}^2 + o + 2o_1 + z$$

$$h = (x_{P_i} - x_{c_i})^2 + (y_{P_i} - y_{c_i})^2 + o + z$$

Next the sub-community representative calculates

$$g = h - o = h = (x_{P_i} - x_{c_i})^2 + (y_{P_i} - y_{c_i})^2 + o + z - o$$

$$g = (x_{P_i} - x_{c_i})^2 + (y_{P_i} - y_{c_i})^2 + z$$

Finally at Step 4, participant P_i calculates distance as follows

$$Distance = \sqrt{(x_{P_i} - x_{c_i})^2 + (y_{P_i} - y_{c_i})^2 + z - z}$$

Theorem 2: Neither participant nor any sub-community representative gets the real location of the other party.

Proof: In SSD protocol, **(1).** both participant and representative can pick imprecise locations from their locally generated cloaks to act as their current locations. Moreover, participant encrypts this location using homomorphic encryption and releases it to each representative, such that, the desired computation can be performed on released location data without knowing the decryption key. Note that none of these representatives are able to get this information about the participant. Another advantage of encrypting these imprecise locations is that it allows participant to publish different imprecise locations from his/her cloak to each representative without the fear of location disclosure, since encryption detains the ability of any malicious parties to correlate these released locations using triangulation or inference tools, that can be used to reveal participant's real location. **(2).** the locations of representatives are preserved during the execution of the protocol, as they also employ cloaks to hide their locations and release imprecise locations for computations. Moreover, they add randomness to any intermediate values like a or b when publishing these values to participants. From (1) and (2), we can conclude that the SDD protocol is secure.

8 Analysis of The Tradeoff Between Privacy and Accuracy

When multiple students submit their profiles to super-peers to obtain referrals about which sub-communities they can join, the tradeoff between privacy level and accuracy of referrals depends on released profiles from all students. Super-peers should be able to operate in profiles with high privacy level without impairment in accuracy of referrals. In this part we employ a formal model to quantify the contradictory goals of protecting privacy while maintaining accuracy. Our aim, it to determine profile releasing strategy for participants based on Nash Equilibrium Point such that any participant cannot benefit in terms of increasing privacy level by unilaterally diverging from that point. In order to prove that, let us consider the following: assume we have a set P of M participants and a set of interests/preferences T available in participants' profiles. Each participant x has a subset of interests/preferences $D_x \subset T$ in his/her profile where we denote private profile as $V_x = (v_{xi}: i \in D_x)$. When the participant engages in recommendation process, he/she

sanitizes private profile to the form $O_x = (o_{xi}: i \in D_x)$ which in general different than V_x but $|V_x| = |O_x|$. Super-peers collect different sanitized profiles from participants denoted by $O = (o_x: x \in P)$ in order to extract sub-communities of similar participants, whereas the private profiles that can be collected without privacy concerns is denoted by $V = (v_x: x \in P)$. Recommendations are done using collaboration between super-peers in order to apply a generic function $r(.)$ on collected sanitized profiles to extract a list of available sub-communities available within this community. Subsequently, super-peers compute referrals based on a similarity score between participant's profiles and each sub-community profile as follows:

$$z_x = \frac{1}{M-1}\sum \left| \forall_{\substack{c \neq x \\ y \in D_{P_u}}} o_{cy} \in P_u \right| . \frac{1}{D_x} \sum_{\substack{i \in D_x \\ y \in D_{P_u}}} S(i,y)|o_{xi}| \qquad (1)$$

Where P_u is the profile of sub-community u containing subset D_{P_u} of interests/preferences. $S(i,y) \in [0,1]$ is the similarity value between the preferences of the participant and sub-community. The final list of sub-communities provided for participant x is denoted by $z_x = (z_{xy}: y \in T \,\&\, y \sim i)$, and it is computed as $z_x = r_x(O) = r_x(o_1, \ldots\ldots, o_m)$. Privacy level for participant x is quantified based on both private and sanitized profiles that can be represented by function $pl_x(V_x\,,\, O_x) = H(V_x) + H(O_x) - 2\big||o_x| - |v_x|\big|$ that reflects the variation of information between private and sanitized profiles. While accuracy of referrals is quantified based on sanitized profiles of all participants in this sub-community, if we assume $\bar{z_x} = r_x(V) = r_x(v_x, \ldots\ldots, o_m)$ is the generated list of sub-communities for participant x if he/she releases his/her private profile, then accuracy can be quantified as $|z_x - \bar{z_x}| \leq \varepsilon \Leftrightarrow |r_x(o_x, \ldots\ldots, o_m) - r_x(v_x, \ldots\ldots, o_m)| \leq \varepsilon$, where ε denotes the upper-bound of maximum convenient error in the extracted sub-communities. Based on equation (1), we can state accuracy as follows:

$$\frac{1}{D_x}\frac{1}{M-1}\sum \left| \forall_{\substack{c \neq x \\ y \in D_{P_u}}} o_{cy} \in P_u \right| . \forall_{\substack{i \in D_x \\ y \in D_{P_u}}} S(i,y)\big||o_{xi}| - |v_{xi}|\big| \leq \varepsilon \qquad (2)$$

Formally, we can define participant's objective when participating in recommendation request as max $pl_x(V_x\,,\, O_x)$ subject to $|r_x(o_x, \ldots\ldots, o_m) - r_x(v_x, \ldots\ldots, o_m)| \leq \varepsilon$. This means that participant x has the choice to release any version of his/her sanitized profile that is based upon his/her private profile. This objective is only defined from participant

x point of view that is assumed to be having a selfish but rational attitude. Formally, we can employ Nash Equilibrium Point to select the optimal version of the sanitized profiles of each participant such that no participant can benefit by choosing different sanitized profile in terms of increasing privacy level. Then for each participant x there holds $pl_x(V_x\,,\,\bar{\bar{O}}_x) \geq max\ \vee_{\bar{O}_x} pl_x(V_x\,,\,\bar{O}_x)\ \forall\ \bar{\bar{O}}_x \neq\ \bar{O}_x$. In order to get a convergence, we run different iteration between participants and super-peers, where for each iteration we sanitize the private profiles of participants using our proposed protocols with different granularities. Then, we release these profiles for recommendation process to illustrate different privacy levels. In the end of each iteration, a linear programming (LP) problem is formulated for each participant as follows $max\ pl_x(V_x\,,O_x^t) = H(V_x) + H(O_x^t) - 2\left||o_{xi}^t| - |v_{xi}|\right|$ subject to

$$\frac{1}{D_x}\frac{1}{M-1}\sum\left|\forall_{\substack{c\neq x\\ y\in D_{P_u}}} o_{cy}^t \in P_u\right|.\forall_{\substack{i\in D_x\\ y\in D_{P_u}}} S(i,y)\left||o_{xi}^t| - |v_{xi}|\right| \leq \varepsilon$$

That includes the released and private profiles for all iterations. This optimization problem is solved by each participant based on the profiles of other participants which have been collected by super-peer during recommendations process (which assume to be fixed in all iteration). If we set $a_{xi} = \left||o_{xi}| - |v_{xi}|\right|$ and, $a_i = (a_{xi} : i \in D_x)$ the problem can be written as follows:

$$\max pl_{xi}\,(v_{xi}\,,o_{xi}) = H(v_{xi}) + H(o_{xi}) - 2a_{xi} \text{ Subject to } b_{xi}.a_{xi} \leq \varepsilon(M-1)$$

$$\text{Such that: } b_{xi} = \frac{1}{D_x}\sum\left|\forall_{\substack{c\neq x\\ y\in D_{P_u}}} o_{cy} \in P_u\right|.\forall_{\substack{i\in D_x\\ y\in D_{P_u}}} S(i,y).$$

The solution to this problem is found among the extreme points of the feasible set:

$$i' = \arg min_{i\in D_x}\frac{b_{xi}}{[H(v_{xi}) + H(o_{xi})]}$$

Then

$$a_{xi'} = H(v_{xi'}) + H(o_{xi'}) - \frac{2\varepsilon(M-1)}{b_{xi}}$$

For other items such that $i' \neq i$, it is $a_{xi} = 0$. from our experiments follows that privacy maximization is achieved because of our careful selection of both of privacy and accuracy metrics.

9 Experiments

In this section, we describe the implementation of our proposed solution. The experiments were run on 2 Intel® machines connected on local network, the lead peer is Intel® Core i7 2.2 GHz with 8 GB RAM and the other is Intel® Core 2 Duo™ 2.4 GHz with 2 GB RAM. We used MySQL as a data storage for the participants' profiles that is acquired by learning agent. The CRS has been implemented and deployed as a web service while the *EMCP* has been deployed as an applet to handles the interactions between its owner, CRS and other participants; it uses the implementation of the MPI communication standard for distributed memory implementation of our proposed protocols to mimic a distributed reliable network of peers. Our proposed protocols are implemented using Java and boundycastel© library, RSA key length is set to 128 for the experimental scenario. The experiments were conducted using a dataset containing preferences and locations data of 6000 students during various exhibitions in one of the universities in North America. Each of those preferences fell into one of several categories: faculty, main course, study/research domains, academic year, region of birth, region of residence, activities, religious beliefs, sexual preferences, specific expertise, previous study history, awards, time tables, and finally their arrival/departure times to campus. To generate the public profiles for region of birth, region of residence, activities and awards categories, we used Google Directories to extract generalized terms in order to replace them. Moreover for study domains, we used the classification system for each research area like (ACM computing classification system). In order to generalize the time table, we classify times into 3 categories [morning, noon, evening]. We utilize precision and recall metrics proposed in [28] to measure privacy and accuracy of the results. The location records with timestamps that indicate student movement in the city, the x coordinate range is between 0 and 24000 and y coordinate is between 0 and approximately 35000. In order for the SDD to work accurately, accurate location should be provided by the students which do not represent a privacy threat as SDD is executed locally. The collected locations from students have accuracy of 10-15 m, which is good enough for running SDD.

In the first experiment, we want to measure the efficiency of our solution with increasing number of sub-communities. We measured execution time in terms of encryption and transmission time for participants' profiles, as we can see form Fig.(4) our solution re-

quires more communication in consequence of distributed design and communication needs for PCF, PSD and SDD protocols. This acceptable overhead is shared among all participants while the benefit is to protect their privacy without hampering referrals quality. The processes of selecting super-peers and generating distributed keys are done once in the setup time before the start of our protocols, so we omit the required time for them. Since super-peers are the bottleneck in our solution, we implement our distrusted computation using ring topology in order to reduce this risk and computation overhead.

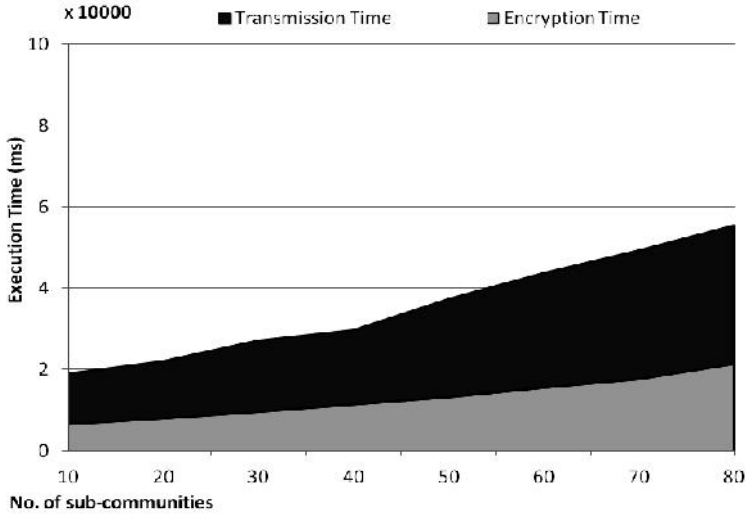


Fig. 4. Execution Time for extracting different sub-communities

In the next experiment, we need to measure the accuracy of extracted sub-communities using CRS service, In order to evaluate the accuracy of our results, we apply density based clustering algorithm called OPTICS [33] to hierarchically cluster students preferences into clusters in a divisive manor. These clusters will be equivalent to sub-communities in CRS, and they are utilized for measuring the accuracy of the produced results. OPTICS algorithm in comparison with an agglomerative method like k-means is capable of detecting clusters with irregular structures which may stand for a sub-community of students. We adapted OPTICS with hybrid similarity measure for short segments of text proposed in [34] to facilitate clustering profiles in plain form. Each cluster represents a sub-community which is constructed from a set of student's private profiles who share the same specific preferences about the same topic. To measure the good-

ness of our results, we considered two error metrics defined in [35] which are grouping error (GR) and critical error (CIE). The first one, the grouping error (GR), takes into account the number of students' profiles included in a sub-community, but belonging to a topic different from the dominant topic in that sub-community. The second one, the critical error (CIR) measures the number of students' profiles belonging to a topic that is not the dominant one in any sub-community. The graphs in Fig.(5) and (6), contain both GR and CIE values for the results obtained from both OPTICS and CRS algorithms for different number of sub-communities. This experiment is performed on two versions of our dataset; students' profiles are generalized then both of PCF and PSD are utilized to discover sub-communities, while OPTICS utilizes students' private profiles directly for clustering process. We can deduce that both GR and CIE for CRS decrease with the increase in number of sub-communities till reaching natural number of sub-communities. This indicates that achieving privacy is feasible and does not severely affect the accuracy of the generated sub-communities.

In the next experiment, we intend to study the effect of increasing the number of released preferences in each recommendation process. Evidently, this increases computation and communication at super-peers, as the whole process is performed collaboratively between these super-peers. Fig (7) depicts that, the increasing the number of released preferences, increases the privacy gain in our system, as the portion of shared preferences between members is increased which will increases the values of precision and recall in our privacy metric. Moreover, the relation between privacy gain while varying the number of participants is presented in the same figure. The privacy gain is independent on the number of participants. As the number of participants increases, the number of released preferences also increases, and so the privacy gain.

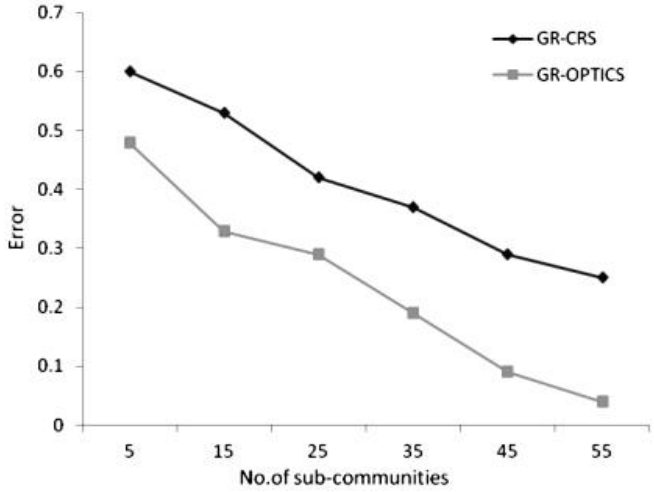


Fig. 5. Grouping Error (GR) of CRS Service

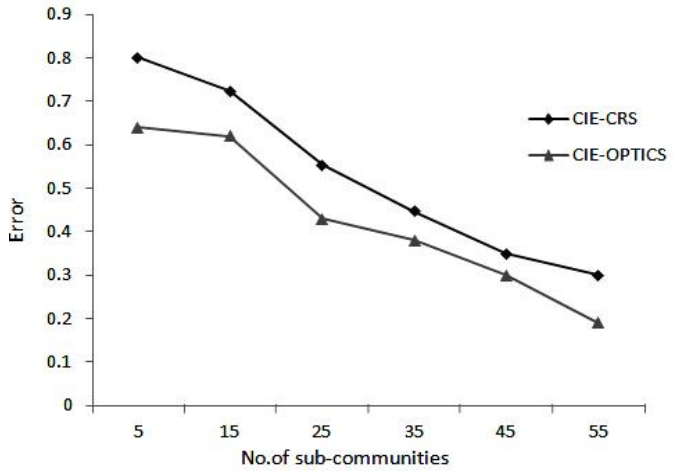


Fig. 6. Critical Error (CIE) of CRS Service

In the next experiment, we intend to measure the accuracy of identifying sub-communities using sanitized profiles in such a way to determine the percentage of sub-community members correctly identified. It is important for our affinity metric to captures similarity values between the same sub-community members such that these values would generally be greater than values obtained for non sub-community members. The results presented in Fig.(8) illustrates that the identified percentage is much higher when privacy level is between 0.1 and 0.3; we can achieve similar results even for privacy level=0.5. We can deduce that CRS is able to identify between 75% and 80% of the sub-

community members with negligible error when the average privacy level was at least 50%.

The following experiment, we measure the impact of adding location data as a part of participants' profiles on the accuracy of generated referrals. Similarity scores between each pair of participants are used by super-peers to compute sub-communities of related participants. When calculating scores, participants take into account two factors, the similarity value between their preferences/interests and distance between their locations. The final similarity scores are weighted and summed based on these two values. Fig.(9) depicts the comparative results of running proposed protocols with varying privacy levels on a dataset containing both preferences and location data. For the same dataset we run this experiment on preferences data only. As it is shown in Fig.(9), the accuracy of referrals generated with preferences data in profiles only achieve higher accuracy than ones with preferences and location data, since the search for suitable sub-communities for referrals will be performed across whole sub-communities representatives. On the other hand, using both preferences and location data in participants' profiles constrains the underlying search space for generating referrals which affect accuracy, since recommendations process will only performed on nearby sub-communities representatives only. However, this does not downgrade the accuracy of provided referrals, since SDD protocol utilizes cloaking strategy around participant and sub-communities locations in order to preserves location privacy, which in turn increases the possibility for each participant to have abundance of nearby sub-communities representatives. Adding location data enhances the quality of provided service with location awareness and guiding facilities and reduces time required to generate referrals.

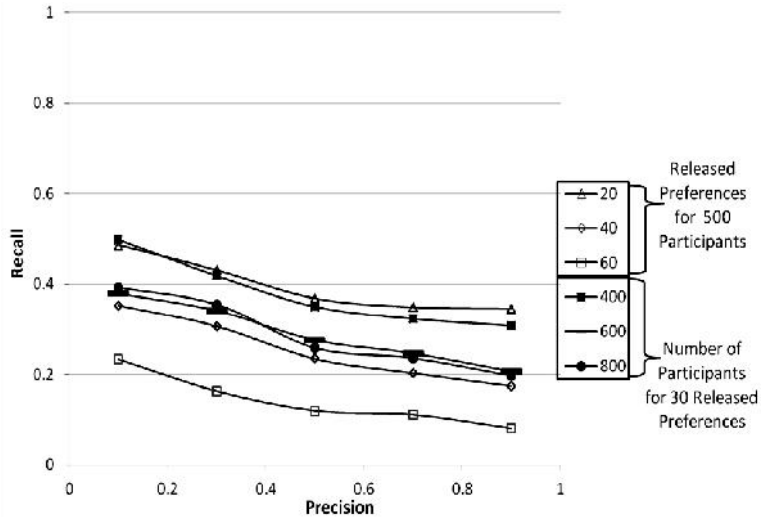


Fig. 7. Privacy Gain with Varying Number of Released Preferences and Participants in Sub-community

In the last two experiments, we evaluated our proposed tradeoff model in terms of attained privacy and accuracy in CRS. We employed the proposed privacy metric presented in Section (7). Fig.(10) illustrates the privacy level for different participants as a function of maximizing the error in the referrals; we notice that as participants lenience to error in the extracted sub-communities increases, the privacy level also increases. Whereas, participants have to reveal more data about their real preferences if they are demanding accurate referrals with less error, which confirms the trade-off between privacy and accuracy of offered referrals. Fig.(11) depicts for different participants, the NEP convergence for the achieved privacy with increasing number of iterations. For each iteration, participants construct sanitized versions of their private profile within various granularities and then they release these sanitized versions to super-peers. Moreover, we wanted to measure the effect of varying upper-bound of maximum convenient error ε in the generated referrals. The achieved privacy with varying convenient error ε is also plotted with increasing number of iterations. It is clear that after a small number of iterations, the NEP convergence is attained for the privacy level. Quick convergence is due to defining our system as a linear programming problem.

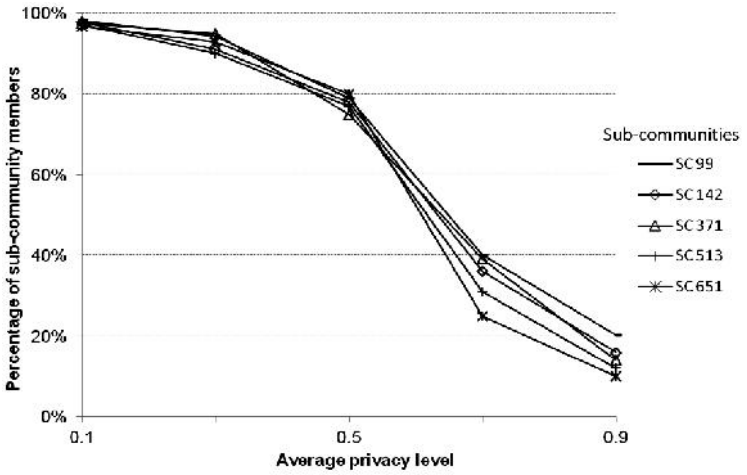


Fig. 8. Percentage of correctly identified members in sub-communities with different privacy levels

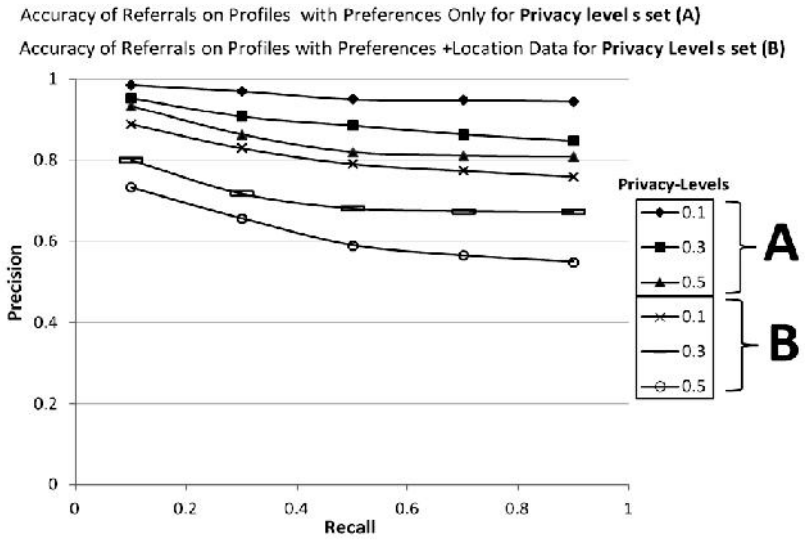


Fig. 9. Resulting accuracy of referrals when combine location+ preferences data with different privacy levels

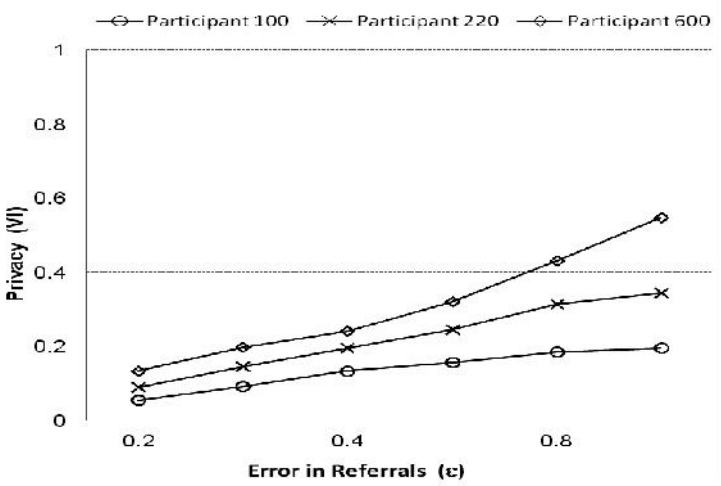


Fig. 10. Privacy achieved Vs. Error in referrals

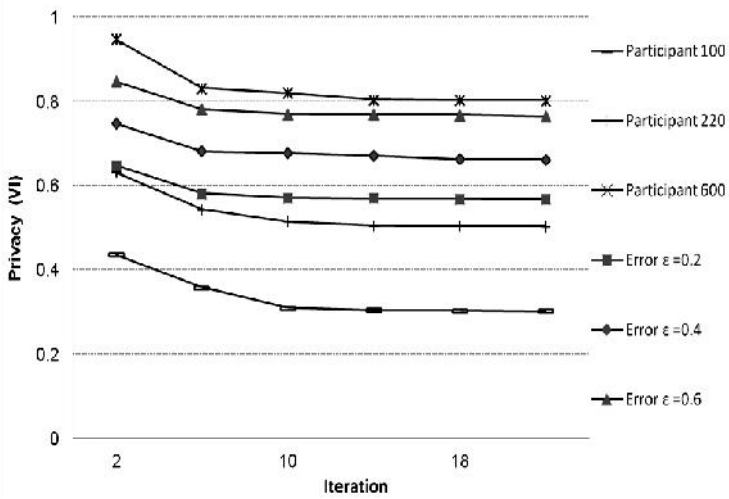


Fig. 11. Convergence for different participants and values of ε

10 Conclusion

In this paper, we presented an attempt to develop an enhanced middleware for collaborative privacy (*EMCP)* for community based recommender service in a university scenario. We gave a brief overview of *EMCP* architecture and proposed related protocols. We outlined our new secure distance detection (SDD) protocol that is utilized to calculate the distance between participants and different sub-communities without revealing their positions. Moreover, we derivate a trade-off model between privacy and accuracy in generat-

ing referrals based on a model that represents the contradictory goals for participants in terms of accuracy and privacy for generated referrals. Based on our experiments, accurate referrals can be obtained with convenient error determined by the participant while attaining convergence for the privacy level. We tested the performance of the proposed protocols on a real dataset. The experimental results show achieving privacy in recommending sub-communities is feasible under our proposed middleware without hampering the accuracy of the recommendations. The future work will include utilizing game theory in order to better define user groups as well as sequential preferences release and its impact on privacy of whole profile.

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2013R1A1A2061978)

References

1. D. Taylor, et al., "Privacy concern and online personalization: The moderating effects of information control and compensation," Electronic Commerce Research, vol. 9, pp. 203-223, 2009/09/01 2009.
2. C. C. Yang, et al., "Visualization of large category map for Internet browsing," Decision Support Systems, vol. 35, pp. 89-102, 2003.
3. K. Petrova and B. Wang, "Location-based services deployment and demand: a roadmap model," Electronic Commerce Research, vol. 11, pp. 5-29, 2011/01/01 2011.
4. F. Xu, et al., "Factors affecting privacy disclosure on social network sites: an integrated model," Electronic Commerce Research, pp. 1-18, 2013/03/26 2013.
5. R. Smith and J. Shao, "Privacy and e-commerce: a consumer-centric perspective," Electronic Commerce Research, vol. 7, pp. 89-116, 2007/06/01 2007.
6. K. D. Mitnick and W. L. Simon, The Art of Deception: Controlling the Human Element of Security: John Wiley \\& Sons, Inc., 2002.
7. J. He, et al., "Inferring privacy information from social networks," presented at the Proceedings of the 4th IEEE international conference on Intelligence and Security Informatics, San Diego, CA, 2006.
8. S. K. S. Cockcroft and P. J. Clutterbuck, "Attitudes towards information privacy," 2001.
9. M. d. Gemmis, et al., "Preference Learning in Recommender Systems," presented at the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD), Slovenia, 2009.
10. F. McSherry and I. Mironov, "Differentially private recommender systems: building privacy into the net," presented at the Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, Paris, France, 2009.
11. P. Koutsabasis, et al., "A descriptive reference framework for the personalisation of e-business applications," Electronic Commerce Research, vol. 8, pp. 173-192, 2008/09/01 2008.

12. A. Esma, "Experimental Demonstration of a Hybrid Privacy-Preserving Recommender System," 2008, pp. 161-170.
13. S. Loureiro, et al., "Secure Data Collection with Updates," Electronic Commerce Research, vol. 1, pp. 119-130, 2001/02/01 2001.
14. J. Canny, "Collaborative filtering with privacy via factor analysis," presented at the Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, Tampere, Finland, 2002.
15. J. Canny, "Collaborative Filtering with Privacy," presented at the Proceedings of the 2002 IEEE Symposium on Security and Privacy, 2002.
16. H. Polat and W. Du, "Privacy-Preserving Collaborative Filtering Using Randomized Perturbation Techniques," presented at the Proceedings of the Third IEEE International Conference on Data Mining, 2003.
17. H. Polat and W. Du, "SVD-based collaborative filtering with privacy," presented at the Proceedings of the 2005 ACM symposium on Applied computing, Santa Fe, New Mexico, 2005.
18. Z. Huang, et al., "Deriving private information from randomized data," presented at the Proceedings of the 2005 ACM SIGMOD international conference on Management of data, Baltimore, Maryland, 2005.
19. H. Kargupta, et al., "On the Privacy Preserving Properties of Random Data Perturbation Techniques," presented at the Proceedings of the Third IEEE International Conference on Data Mining, 2003.
20. B. N. Miller, et al., "PocketLens: Toward a personal recommender system," ACM Trans. Inf. Syst., vol. 22, pp. 437-476, 2004.
21. G. Swamynathan, et al., "The design of a reliable reputation system," Electronic Commerce Research, vol. 10, pp. 239-270, 2010/12/01 2010.
22. A. Elmisery and D. Botvich, "Multi-agent based middleware for protecting privacy in IPTV content recommender services," Multimedia Tools and Applications, pp. 1-27, 2012/03/01 2012.
23. A. Elmisery and D. Botvich, "Privacy Aware Recommender Service using Multi-agent Middleware- an IPTV Network Scenario," Informatica, vol. 36, 2012.
24. A. Elmisery and D. Botvich, "Enhanced Middleware for Collaborative Privacy in IPTV Recommender Services " Journal of Convergence, vol. 2, p. 10, 2011.
25. A. Elmisery and D. Botvich, "An Agent Based Middleware for Privacy Aware Recommender Systems in IPTV Networks," in 3rd International Conference on Intelligent Decision Technologies University of Piraeus, Greece, 2011.
26. A. Elmisery and D. Botvich, "Agent Based Middleware for Maintaining User Privacy in IPTV Recommender Services," in 3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems, Aalborg, Denmark, 2011.
27. A. Elmisery and D. Botvich, "Privacy Aware Recommender Service for IPTV Networks," in 5th FTRA/IEEE International Conference on Multimedia and Ubiquitous Engineering, Crete, Greece, 2011.
28. A. M. Elmisery, et al., "Privacy Aware Community based Recommender Service for Conferences Attendees," in 16th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems, San Sebastian, Spain, 2012, pp. 519-531.
29. P. Paillier, "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes."
30. J. Li and J. B. Pendry. (2008, Hiding Under the Carpet: a New Strategy for Cloaking. Available: http://arxiv.org/abs/0806.4396
31. J. Ram, et al., "A secure multidimensional point inclusion protocol," presented at the Proceedings of the 9th workshop on Multimedia & security, Dallas, Texas, USA, 2007.
32. T. Thomas, "A secure multidimensional point inclusion protocol," CoRR, pp. 109-120, 2007.

33. M. Ankerst, et al., "OPTICS: ordering points to identify the clustering structure," presented at the Proceedings of the 1999 ACM SIGMOD international conference on Management of data, Philadelphia, Pennsylvania, United States, 1999.
34. D. Metzler, et al., "Similarity measures for short segments of text," presented at the Proceedings of the 29th European conference on IR research, Rome, Italy, 2007.
35. D. Cuesta-Frau, et al., "Clustering of electrocardiograph signals in computer-aided Holter analysis," Computer methods and programs in biomedicine, vol. 72, pp. 179-196, 2003.

Appendix E: Pervasive Healthcare Service Health System Scenario

Article XIV

Pervasive Computing Support in the Transition towards Personalised Health Systems

Martín Serrano, Ahmed M. Elmisery, Mícheál Ó. Foghlú, Willie Donnelly, Cristiano Storni, Mikael Fernström

In the IGI International Journal of E-Health and Medical Communications (IJEHMC), Volume 2, Issue 3, July 2011.

International Journal of E-Health and Medical Communications, 2(3), 31-47, July-September 2011 31

Pervasive Computing Support in the Transition towards Personalised Health Systems

Martín Serrano, Waterford Institute of Technology, Ireland

Ahmed Elmisery, Waterford Institute of Technology, Ireland

Mícheál Ó Foghlú, Waterford Institute of Technology, Ireland

Willie Donnelly, Waterford Institute of Technology, Ireland

Cristiano Storni, University of Limerick, Ireland

Mikael Fernström, University of Limerick, Ireland

ABSTRACT

This paper discusses pervasive computing work in the transition from traditional health care programs to personalised health systems (pHealth). A chronological guided transition survey is discussed to highlight trends in medicine describing their most recent developments about health care systems. Future trends in this interdisciplinary techno-medical area are described as research goals. Particularly, research and technological efforts concerning ICT's and pervasive computing in healthcare and medical applications are presented to identify systems requirements supporting secure and reliable networks and services. The main objectives are to summarise both the pHealth systems requirements providing end-user applications and the necessary pervasive computing support to interconnect device-based health care applications and distributed information data systems in secure and reliable forms, highlighting the role pervasive computing plays in this process. A generic personalised healthcare scheme is introduced to provide guidance in the transition and can be used for multiple medical and health applications. An example is briefly introduced by using the generic scheme proposed.

Keywords**:** *Computer Science, eHealth, Emerging Communication Technologies, Healthcare, Personalised Health Systems, Pervasive Computing, pHealth*

1. INTRODUCTION

Traditionally healthcare programs are conceptualized within an isolated vision in what a program can do to support a particular sector in the society. This vision generates the inherent feature that the healthcare management programs itself are responsible of people leaving aside by their specific health requirements. Healthcare programs have acquired a particular

DOI: 10.4018/jehmc.2011070102

interest and evolved in last decades as result of immersion and technological advances in the so-called digital era (eHealth) (European Commission, 2009).

Indistinctly called, eHealth or health informatics has opened, in fact, a new world of applications with multiple benefits in the medical sector. This advance is mainly result in the evolution of more powerful devices and with increasing processing capacities, efficiency and energy consumption. The difference between named in one or other form founds on the applications and implemented devices, as we will discuss later in this paper.

Systems deployment and implementation support also plays a decisive role to classify chronologically such health programs and/or systems evolution according with their health services and applications. Digital health (Gatzoulis & Iakovidis, 2008) emerges when the advances informatics are visualized as the tool to support health programs, to facilitate activity and improve services, typically and mainly for processing huge quantity of records and health patters that before was almost impossible to do so.

In last two decades the technological development has suffered exponential advances and research activities for the application of Information and Communication Technologies (ICT) in healthcare sector has advanced enormously. These advances hampering interdisciplinary efforts to establish networks and tools assisting health care professionals (MobilHealth, 2010; HealthService24, 2010; Continua, 2010). This movement of research ICT support, where technological resources are directed towards healthcare and so called eHealth (European Commission, 2009) is every day attracting not only the attention of the scientific community else industry sectors where new products and services can emerge and explored as well.

In the last decade, late 1990s, the focus in health shifted, generic health programs have to be narrowed and use the advantages of the technological expansion and the development that device's capabilities offers. Recently research in health supported by ICT's expands widely the service possibilities, this time not only particular people sectors are recipients. Likewise in recent years an adoption of new culture in the society has changed, with virtual social models expansion, a new vision person-centric in healthcare emerged. This person-centric view is result of a more knowledge-based culture (Giustini, 2006), health sector is not an exception in this immersion and people have modified their behaviour patterns and culture towards the health. All these changes and its consequences are important to be documented, mainly as reference for future implementations in medical and health solutions. In this paper we summarise both the pHealth systems requirements providing end-user applications and the necessary pervasive computing support to interconnect in secure and reliable forms device-based health care applications and distributed information data systems, highlighting the role pervasive computing plays in this processes, secure in terms of privacy of the health data record and reliable in terms of technological support to interconnects health applications offering privacy.

The rest of the paper is organized as follow: Section 2 describes related work offering a guided transition from traditional health programs to pHealth systems. Section 3 discusses the personal healthcare systems, and then introduces some research challenges. Section 4 describes the Role of Pervasive Computing Supporting Personalised Healthcare Schemes to identify both the importance of services personalisation in healthcare applications and the necessary pervasive computing support for processing data and interconnects health applications. Section 5 introduces our vision for a generic personalized health care scheme as tool to promote standard health programs based on pervasive computing services support as well as integrated information exchange models. Section 6 introduces our vision for generic implementations as practical tools following the proposed personalised health scheme describing and ECG data scenario as an example. Section 7 summarizes the research advances and concludes this paper.

International Journal of E-Health and Medical Communications, 2(3), 31-47, July-September 2011 33

2. HEALTHCARE SYSTEMS RELATED WORK

This section is devoted to describe briefly research and implementation efforts pointing out to those approaches that pave the way to overcome technological limitations that hinder the introduction of ICT's to support healthcare programs and facilitating the development of pHealth systems (Serrano et al., 2010; Borrel, 1993).

Trends in next generation of networks and systems related with medicine and healthcare demand applications that would shape the requirement to allow prevention of diseases even before they are apparent by using assisted sensors and networks (Zhu et al., 2009; Lupu et al., 2008). To achieve this objective the personalisation of services is crucial in personalised healthcare systems (pHealth) (Gatzoulis & Iakovidis, 2008)

Multiple advantages arise when computing systems are being used to, for example assist medical professionals to analyse health historical records and exhaustive distributed data analysis about diverse patient's health status. The personalisation of healthcare services is an example of such advantages (Neves et al., 2008a). With personalised healthcare services people can receive more accurate diagnostics and early medical assistance even before diseases are apparent for instance. pHealth systems demands about personal devices computing capabilities to locally and remotely process information as well as network infrastructure high performance (ICT's) able to react in real-time to variations in the information being collected, processed and storied remotely for patient about health levels.

As an example of end user's device, applications have been developed to create an easy interactive human tool in a form of easy and accessible health care interface (Neves et al., 2008b). In the other hand the ICT support for phealth system is conducted by information service and data management systems. Particular technologies for management and formal representation of health information are some critical techniques in how ICT mechanisms are used to satisfy information systems requirements and conduct the future research and implementation in pHealth systems activity.

pHealth systems design involves an end user participatory model where the proactiveness of patients is considered more seriously than ever before. Advances in end user's devices promoting dynamic patient participation are widely implemented and does not represent a technological limitation anymore, there are some good guidance to follow in this sense when systems are designed (European Commission, 2010). System applications for medical education of both GP's and patients are another example where end user participation is considered as critical role in the system design (Sandars & Haythornthwaire, 2007; Crespo, 2007). A more dynamic participation in self-help and self-management discussions about health and diseases demonstrates how people are addressing participation in an evolving technological systems advance (Preece, 1998). This important change of participative culture is reflected when people self-managing a particular disease by connecting to an on-line community (Ferguson, 2010).

The notion of patient empowerment has recently gained much attention especially in the context of chronic disease analysis where tracking records must be collected and studied for long periods (Wagner et al., 2010; Barlow et al., 2010) (Bodenheimer et al., 2002). This currently quite popular notion is recently found in medical literature especially in the field of nursing and patient research of chronic disease (Cahill, 1996; Trummer et al., 2006), and of qualitative health research (Opie 1998; Mol, 2008). In different ways, all these studies point to the fact that the traditional biomedical model often act as a barrier to real patient empowerment, agency and agenda. For more details and evidence on how medical model fail to address the patient agenda in the doctor-patient interaction, refer to Levenstein et al. (1986) and Barry et al. (2000).

The design of a generic personalised healthcare scheme poses a common challenge personalizing health and information systems to

satisfy the increasing phealth systems demands. In this sense the web role is also crucial. As an example the web is witnessing the rise of patients' blogs, for example taking a specific case of diabetes disease. Multiple life sites exist and many others will arise as much as the people use the internet and technological tool as daily activities controlling their diseases (e.g. diabetesmine.com, assertivepatient.com). Another example of this, dynamic people participation, can be found in health social networks (e.g. dailystrengh.com, patientslikeme.com), or health wikis (e.g. Fluwikie.com; en.wikipedia.org/wiki/health), Health UGC site (e.g. drugs.com), health podcasts (e.g. podcasthealth.com) and video sharing (e.g. health channels in youtube.com or ivillage.com, iheathtube.com, icyou.com). In this paper we are not evaluating the quality of the information channels we are only evaluating people participation which is increasing every day towards a more health knowledge with self-management orientation, thus pHealth systems will acquire a more direct impact in the people life. Taking as example the quantity of people which diabetes is a disease (ECG), some applications that have attracted the attention in this study are those developed to Iphones (e.g. Sensei My Diabetes Guide, Glucose Charter, DiabetesPilot, OS 3.0 Life Scan, etc.) so this kind of end-user applications follows a human computer interaction philosophy (HCI).

In the other hand ICT's efforts have been developed and studied for implementing necessary infrastructure for pHealth support. These studies are being conducted in two levels, the first oriented to the sensor area to collect and managing all the information (Callaway & Callaway, 2003). Approaches for personal monitoring (Montón et al., 2008) and the second which principally is oriented for offering a more flexible, easy and accessible set of tools able to handle the health-profiles information (Gray & Salber, 2001; Judd & Steenkiste, 2003). These infrastructures must offer security in terms of privacy and efficiency when the information is being used to support more complex information mechanisms and systems to, for example, inform patient health profile, or send patient health status and then react when an emergency occurs.

The Ubiquitous computing paradigm rely on the use of non-invasive monitoring approaches through wearable computing smart wristbands or body-sensor networking and pervasive computing, including tracking of everyday interactions with wireless sensors and RFID-tagged devices.

Mostly tele-care projects design largely rely on widely clinical aspects by talking to a clinical audience, this is not anymore the objective when personalized health systems are designed, this issues includes problems having too many control variables and the difficulty in calculating number of participants. So we have to discuss on how current evaluation techniques privilege a clinical perspective, and underlines the need to integrate social shaping approaches to rebalance normative approaches in assessing complex interventions in personalized health care systems (Storni, 2010).

Diverse initiatives regarding context composition (Buchholz et al., 2004), service management (Serrano et al., 2007), ontology engineering, Autonomic communications, to create an appropriate personalised health service supported by an architectural framework focuses specifically on the issue of a self-managing end-user and service driven context-resource orchestration (Davy et al., 2007; Hochstatter et al., 2007), in pervasive computing. The generic pHealth systems proposed in this paper seeks for automate the information-resource processes via an open and secure control that makes decisions using knowledge supported by actions and operations generated based on health patient information and health profiles data models.

3. HEALTHCARE TRANSITION TOWARDS PHEALTH SYSTEMS: CHALLENGES

Healthcare has always played a crucial role in the evolution and development of the societies. Traditionally, generalized tendencies of use

International Journal of E-Health and Medical Communications, 2(3), 31-47, July-September 2011 35

populations to assume that medical programs must be equal to each person exist (Conrad, 1985) or at least the health programs are to be implemented in this form.

Traditional healthcare programs can do to support to a particular sector in the society which, is unacceptable in current digital and technological era (European Commission, 2009). This traditional vision generates the inherent feature that the healthcare programs itself are responsible of people leaving aside by their specific health requirements. So, for example, traditional healthcare programs affect a sector when non-correct scheme attention is provided or even when unknown patters of a disease are detected. As result people is simply leave out of the scheme. Healthcare programs acquired a particular interest and evolved in the last decades as result of the immersion and the advances in the so-called digital era, this immersion is result of the interest to solve, between many others, the above-mentioned problem.

The variety of healthcare programs establishes an unequal evolution of health in the societies and erroneously when health programs are not planned correctly the responsibility of the health is delegated to assurance a minimum of health care guarantees for the population being served, which society are unequal served. To face up to this challenge, medical science and health care programs have improving programs constantly. Today, with more research advances, new social models and their easy adoption: and as result of constant evolution in technology immersion, personalisation in medical applications development has been benefited and more easily implemented.

This new vision also known as pHealth defines complete personalised medical support and technological assistance (Zhu et al., 2009; Lupu et al., 2008). The implementation of pHealth programs is independent to each society and the efficiency in its application depends of social, politics and economics factors. pHealth defines complete personalised medical support and technological assistance which are discussed along in this paper. This pHealth scenario demands devices with high and powerful computing capabilities to locally process information while in the other hand complex information and communication support systems (ICT's) able to react in real-time to variations in the remotely information being collected about patient's health levels is necessary. As an additional key challenge the information being monitored remotely must be handled in a private and reliable form as result in the security demand above the nature of the information being managed serving as social healthcare networks.

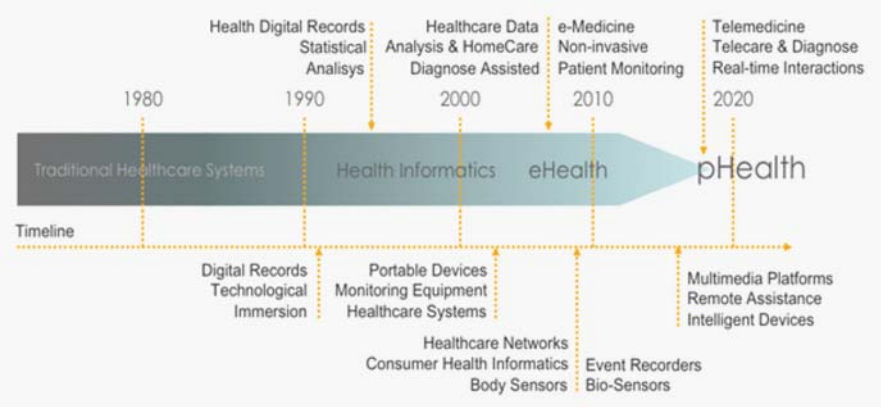
Health informatics or medical informatics is the intersection of information science, medicine and health care. It deals with the resources, devices and methods required to optimize the acquisition, storage, retrieval and use of information in health and biomedicine (Mantzana & Themistocleus, 2004). Health informatics tools include not only computers but also clinical guidelines, formal medical terminologies, and information and communication systems.

eHealth (also written e-health) is a relatively recent term for healthcare practice which is supported by information processes and communications technologies. The term is inconsistently used: some would argue it is interchangeable with health care informatics and a sub set of health informatics, while others use it in the narrower sense of healthcare practice using the Internet.

However recent advances in technology and medicine are, and will continue, hampering important evolution in the health care programs. Personalised healthcare scenarios (pHealth) require increased contents interoperability and the fully integration of the user's health information from sensors, systems and networks (Arriola et al., 2008; Herzog, 2006).

Figure 1 summarizes healthcare programs evolution and their most important services and associated technology platforms or devices. The implementation or not of such programs is independent to each society and the efficiency in its application depends of social, politics and economics factors, but generally a health culture provides better results.

Figure 1. From healthcare to personalised health systems



pHealth can encompass a range of services that are at the edge of medicine/healthcare and information technologies (Della Mea, 2001; ITU, 2008):

- Digital Medical Records: to enable communication of patient data between different healthcare professionals (GPs, specialists, care team, pharmacy) in a secure form. It is featured by statistical analysis of the health data records.
- Consumer Health Information: as information necessary for end-user orientation and informing on medical topics, this information could be electronic text or video-based.
- Mobile Health (mHealth): Service featured by the using mobile devices in collecting aggregate and patient level health data, providing healthcare information to practitioners, researchers, and patients, real-time monitoring of patient vitals, and direct provision of care.
- Healthcare Information Systems: Software and Infrastructure solutions for managing health profiles, appointment scheduling, patient data management, work schedule management and other administrative tasks surrounding health.
- Medical Networks: This services goal is providing powerful computing and data management capabilities to handle large amounts of heterogeneous data for accurate prescriptions particularly in non-common health profiles.
- Telemedicine: Measurements, Diagnose and treatments with not require patient movements. A feature of this service is avoiding patients travelling to visit a specialist or facilitate specialist job when larger catchment distance.

4. PERVASIVE COMPUTING SUPPORTING PERSONALISED HEALTHCARE SCHEMES

As main objective in this research summary section is to identify both the importance of services personalisation in healthcare applications and the necessary pervasive computing /communications support for processing data and interconnects health systems/applications. pHealth systems pose several technological challenges; some of them are listed and described. A brief description of methods and mechanisms serving as guideline in our approach to introduce a generic personalised healthcare scheme are included in this section.

- Information sharing – use of standards

International Journal of E-Health and Medical Communications, 2(3), 31-47, July-September 2011 37

Particularly in this research we are using Health Level Seven International (HL7). HL7 is the global authority on standards for interoperability of health information technology. HL7's vision is to act as the best and most widely standards in healthcare. In the other side standards for information sharing protocols include extensible markup language (XML), simple object access protocol (SOAP), and web services languages (WSDL).

- Real-time communications – high capacity processing

End users need to exchange information instantly or with negligible latency. Real-time communications (RTC) are necessaries to establish unified interfaces to facilitate efficient, convenient communication among users in a business environment. Stand-alone RTC protocols are often customized to meet requirements of specific operations. Thus RTC data may be embedded in common business applications and be an alternative supporting information exchange. Real-time communications are necessary as enablers and/or facilitators to update information efficiently.

- Networking management – autonomic solutions

Activities, methods and procedures regarding the operation, administration, maintenance, and provisioning of networked systems. Operation deals with keeping the network up and running smoothly. Administration deals with keeping track of resources in the network and how they are assigned. Maintenance is concerned with performing repairs and upgrades. Provisioning is concerned with configuring resources in the network to support a given service.

- Intra-domain communications – information modelling

Data and information sharing imply modelling, representation, and exchange by using data and information systems with the objective of sharing information to provide added-value services. One traditional inter-domain model is the invocation; which is used to allow an arbitrary interface to be used by other domains.

- Synchronization – interoperability and inter-domain

Refers to the idea of keeping consistent multiple copies of a dataset with other ones, or to maintain data integrity along a distributed data system. Process synchronization or serialization are those application used by particular mechanisms to ensure that two concurrently-executing data sets do not differ.

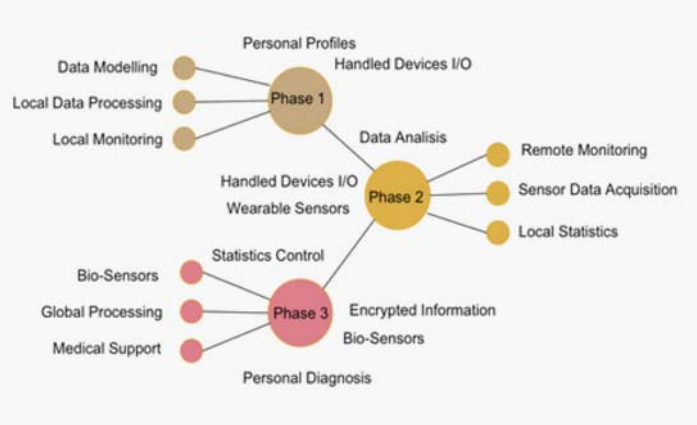
- Trust and security – reliability and deployment

Trust and security are based on a philosophy of decentralizing decisions, and as consequence of this, the creation of open and decentralized systems and stable and secure services are required. In current service management systems and pervasive computing solutions is crucial to protect the system and its sub-systems. Trust is broadly accepted as required for modelling, analysing, and managing decisions within certain security levels.

- Privacy – sensitive Information

One of the biggest problems when healthcare information is being handled is the privacy. Sensitive information need to remain hidden, actually encryption is used as mechanism to achieve privacy levels required. By the nature of healthcare information it is crucial controlling what information one reveals about oneself over the data and information systems and the Internet, and important as well result to control who can access that information.

Figure 2. Pervasive computing in personalised health systems



5. A GENERIC PERSONALISED HEALTHCARE SCHEME

Designing personalised health care programs according to individual requirements and based on information from people's wearable sensors and easy-handy daily use devices and integrating health monitoring systems in people activity are ways to obtain health status everywhere people moves.

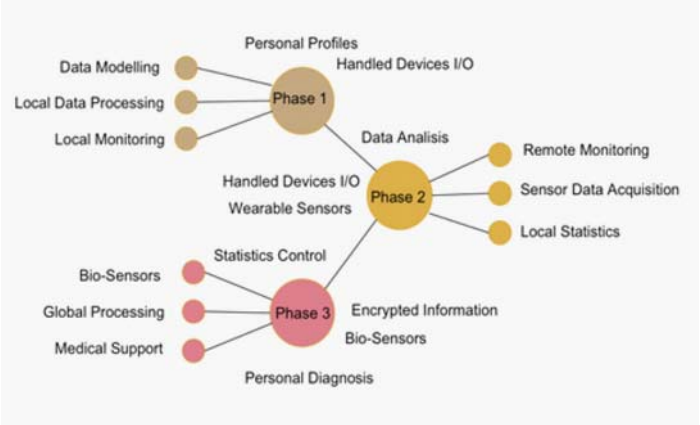
The design of personalised healthcare schemes must addressing research and technological efforts under the generic objective of supporting society, but with the challenge to be adaptive enough to each person requirements and/or necessities. One of the key goals is addressing the above personalised healthcare program scheme with ICT support. The main objective in this section is to provide guidance when designing personalised end-user applications and the necessary ICT support to interconnect in a secure and reliable form, personal health applications with distributed information data systems. In Figure 1 this scenario is depicted, we observe some interaction between users (patients) and professionals.

As depicted in Figure 2, we observe features and requirements in the information and in technology capacities as well, examples of pHealth applications is to inform medical professionals of current health status about patients and then patients can receive earliest possible medical assistance before diseases are apparent, another example is the remote monitoring and tracking systems based on specific health metrics following the process of diseases or post-intervention progress. Collaborative and interdisciplinary research activities are conducted to provide proof of this concept with respective technological ICT support (Serrano et al., 2010).

A generic personalised health care scheme must consider and envisage personalized healthcare systems with ICT systems like an inherent synergy between end-users and technology where the end user play a dynamic roll over the constitution of the health information. So network building up wide decentralized information network based on end-user mobility must be as one of main goals.

In a generic personalised health care scheme, development of patient centred systems is crucial and it highlights the patient role at centre of healthcare research and technological development practices. This feature imply the correct understanding and daily patient dealing with medical information, prescription, medication and the interaction as well as use of health-care technology. As result the users (patients) activate local interactions with more or less institutionalised actors (professionals), often walking around official channels and

Figure 3. Proposed pHealth research model



may develop non-compliant strategies to prevent their diseases taking undue control over their lives.

A dynamic patient's role is then frame worked rather than presuppose patients as passive actors within the healthcare system, this feature acts as a playground in the pHealth systems for developing new applications and services. Autonomous patients will develop their own medication practice, according with personal assumptions based on health profile and symptoms, even though if it may not coincide with the institutionalized health professionals recommendations and even prescriptions (Serrano et al., 2010; Borrel, 1993; Conrad, 1985). In fact this autonomous behaviour could be seen as non-compliance, however from the patient's perspective it is not according their feelings.

In this generic healthcare scheme we assume patients pro-actively build local knowledge in order to deal with the practicalities and solution alternatives of their health problems.

Thus this information can help to enrich health knowledge data and contributes to research activities. For example if a drug is being consumed by mid-long term period of time the monitoring of side-effects are many times difficult and expensive to track in order to improve or change the drug, but if the patients play the role of self-monitoring assisted with ICT support, medical and professional assistance, the benefits in the patient's health and feedback about effects of the medication is crucial for developing new medication schemas or developing new drugs.

We propose a three-phase research to achieve the pHealth system model depicted in Figure 3. Here after a description about each phase. This generic model scheme is applied to any transition model from traditional healthcare programs into a personalised deployable system.

Phase 1 is featured mainly by end user handled devices applications offering local analysis and local statistics about health levels. In this way a self-management control to health's patient levels and a more self-care culture increase people's conscientious about health. In this phase, ICT infrastructure provides communications support offering temporally remote monitoring when user's devices are synchronized at home and office as part of the access to the central health data infrastructure.

Phase 2 concentrates in network reaction based on relevant health statistics and levels, for example, if the local analysis about health levels for glucose or heart rate reflects a strong variation and an abnormal tendency the ICT infrastructure will be able to advise to the end user and at the same time dial for general practitioners (GP's) assistance and adequate advisement to prevent mayor problems or simply to visit a hospital for a more detailed medical

Table 1. Generic personalised healthcare scheme - research phases

Research Phase	Data Acquisition Method	Health Data Use	ICT Infrastructure	Applications
Phase 1	Patients Input	Local Analysis Local Statistics	Communications Support	Remote Monitoring Home and Office Access Control
	Handled Devices I/O			
	Technology Support			
Phase 2	Sensor Data Acquisition	Local Analysis Regional Analysis Local Statistics	Network Reaction Based on Relevant Health Information	Remote Monitoring Home, Office and Eventually Specific Hospitals
	Handled Devices I/O			
	Technology Support			
Phase 3	Wearable and Bio-Sensors Acquisition	Local & Global Analysis Local & Global Statistics	Network Adaptive Process Based on Health Information	Remote Monitoring and World-wide Access for Professionals.
	Handled Devices I/O			
	Technology Support			

revision. In this phase sensor data acquisition is considered as feature necessary to support the personalised health scheme. Both Handled devices statistics and sensor data acquisition combines a local analysis which is used to provide valuable regional analysis when the patient visits the hospital for example. This phase is focusing in people with already detected symptoms and diseases, which additionally to local analysis and local statistics need tracking and more professional health assistance.

Phase 3 concentrates to integrate wearable and bio-sensors data acquisition connected locally with handled or ad-hoc design devices to provide local and global statistics. Ideally world-wide information data is provided to support remote monitoring and professional assistance to for example, more accurate diagnosis based on tracking health care profiles and other previous disease experiences. In this phase ICT infrastructure provides network adaptive communications support to, for example, send instant messages to relatives if some emergency occurs or establish emergency calls to medical assistance or requesting medical support. This phase focuses in people with non-detected symptoms and diseases. Table 1 summarize and shows the proposed three main phases.

6. TEST AND EXPERIMENTATION: AN EXAMPLE

As described along this paper designing personalised health care programs according to individual requirements is a complex process. A premise on these complex health care systems design is that distributed information from people's health status (wearable sensors) and easy-handy daily use devices dedicated on health monitoring during normal levels of people activity must be to obtain everywhere people moves (Neves et al., 2008a).

A) Data Acquisition and ICT Infrastructure

This scenario demands capabilities for analysis distributed information and locally process it to afterwards be compared and processed remotely (Herzog, 2006). Cluster Computing and other software computing techniques have demonstrated their advantages to support this feature offering scalability as well as availability in the collected and processed data (Neves et al., 2008b). However is clear traffic or availability assurance increases and security and privacy in the information is extremely necessary, so

efficient algorithms when health data is being processed must be developed considering these security and privacy requirements.

Privacy is an important issue for different user; it prevents the wide spread of various useful services on the internet (Cranor, 1999). In this paper we briefly discuss a scenario for a healthcare provider performs remote diagnosis by analyzing the ECG patience's signs (Elmisery & Fu, 2010) to assign (diagnose) cardiopathy condition as an example, however any other disease or health records can be used. As demonstrative example we aim to address privacy issues.

The process is accomplished using clustering of ECG signals. Our aim is to permit clustering of ECG signals without learning any private information about the patient or ECG signal. ECG signal allow the untrusted providers to infer different mental and physiological conditions for the patients (depressed, afraid, walking or running, etc.).

The user will have the choice to store the resulting analysis to his medical health record or simply omit it. If the user decides to store it he should provide the system with his identification data. In order to control the privacy level, the input signal is transformed to wavelet domain that is effective in concealing the ECG signal. The data records (ECG signals) are decomposed in a way for achieving a tradeoff between accuracy and privacy.

Modern medicine can benefits from cluster computing by using user's health information profiles offering secure and personalised support to, for example, provides early assistance, quick response when symptomatic diseases are detected by local and remote monitoring analysis, provides more accurate diagnostics to the patients and also provides support for monitoring progress of diseases as well as intervention and therapeutic post intervention procedures.

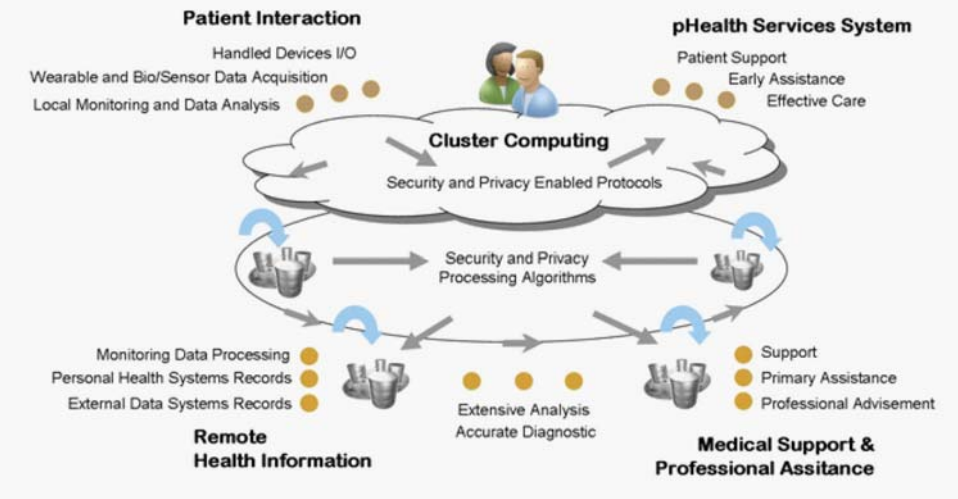
B) Health Data Use and Applications

We envisage supporting pHealth systems with Cluster Computing like an inherent synergy between software systems and technology where the end user (i.e. patient and/or healthy people) play a dynamic roll over the constitution of the health information data base and network to constitute a wide decentralized information network supported by cluster computing applications and management systems based on end-user mobility demands. The development of patient centred systems is crucial and it highlights the patient role at centre of healthcare research and technological development practices. This feature implies the correct understanding and daily patient dealing with medical information, prescription, medication and the interaction as well as use of health-care technology.

In other words, cluster computing assist personalised health systems to support healthcare schemes offering multiple services by using people's health profiles. Additionally a key goal when cluster computing supporting personalised healthcare system (pHealth) is people activity/freedom is not affected as the analysis is based on data is taken during normal activities is being conducted or regular-based medical checks. Modern medicine demands pHealth programs allowing people continue on the move and enable self-managing health schemas to envisage a more dynamic and interactive in real-time patient-doctor information exchange. A clear advantage when using this dynamic exchange is for example assist professionals emitting more accurate diagnostics and tracks patient's health levels.

In Figure 4, we depict our current research approach and how we concentrate on cluster computing management addressing security and privacy in the heath data record to support healthcare systems from a perspective of personalised healthcare data management application. In this scenario our aim is to introduce a data analysis clustering algorithm in diagnosis applications for cardio-vascular signals, this data analysis study and test result are outside the scope of this paper and by space restrictions we are only making mention to our current and other previous research work (Serrano et al., 2009a).

Figure 4. Cluster computing assisting healthcare systems



As mentioned along this paper "A dynamic patient's role is then frame worked rather than presuppose patients as passive actors within the healthcare system"; this feature acts as a playground in the pHealth systems for developing new applications and services.

It is a common behaviour to find autonomous patients following their own medication practice, according with personal assumptions based on health profile and symptoms, even though if it may not coincide with institutionalized health professionals recommendations and even prescriptions (Borrel, 1993).

In fact every day more and more people rely on logging data in electronic forms and receive, if not instantaneously, daily basis feedbacks about their health status rather than wait until disease symptoms are apparent, o a medicine professional emit a diagnosis, and this is a change on traditional health system culture.

People and patients act pro-actively building their health knowledge in order to deal with the practicalities and even search themselves solution and/or alternatives of their health problems. Thus this self-build knowledge about health can help to enrich health knowledge data and contributes to research activities about health care impacting society directly (Serrano et al. 2009b). For example in the example of a new drug is being consumed by mid-long term period of time the monitoring of side-effects many times is very difficult and expensive to track in order to improve or change the drug, and assuming the patients play the role of self-monitoring assisted with the adequate ICT's support, here is where cluster systems can provide data enabling medical and professional assistance and continue con the phase 2 and 3 in the proposed generic personalized healthcare scheme.

7. CONCLUSION AND FUTURE WORK

The research and development of pHealth systems is intensive and significant advances become to be reality as result of the ICT's support, this interdisciplinary activity using more extensive technological platforms promotes that people engage day by day and more easily to a culture of self-management in their health care diagnose and treatment processes.

This paper represents research guidance for developing the Personalised Healthcare Scheme by Research Phases introduced. This model

comply with the requirement of immerse devices in daily patients activities with not necessary to change life activity to be attended, receive medical assistance and or simply health patterns monitoring. Patients are only requested to be at hospitals or specialized medical centres when it is absolutely necessary, as result of measures informed constantly.

Our pHealth scheme design involves end user participatory model where the pro-activeness of patients is considered more seriously than ever before. In this approach technology advances in end user's devices promoting dynamic patient participation does not represent a technological limitation anymore. These and many other examples are just some of the first instances of people's tendency to acquire a self-management healthcare culture. We are conducting research for ICT's support in this sense as well.

This paper has introduced our vision for cluster computing supporting personalised health systems, particularly addressing privacy. As an illustrative example of this approach, secure heart rate monitoring of signals (ECGs) has been briefly discussed.

Research efforts have been conducted to promote cluster algorithms as an alternate privacy solution for finding out data similarities between particular cardio-vascular pattern and cardio-vascular patterns from patients with problems previously diagnosed/detected.

We will continue investigating pervasive computing techniques to map patterns with the objective of support secure and privacy levels for managing healthcare patterns and offer support in diagnostics of diseases and produce reactive solutions in the communications systems.

ACKNOWLEDGMENT

This research activity is being funded by High Education Authority (HEA) into the PRTLI Cycle 4 research program in the framework of the project Serving Society: Management of Future Communications Networks and Services.

REFERENCES

Arriola, A., Brebels, S., Valderas, D., Blasco, J. M., Hernández, J. F., & Montón, E. (2008, May 21-23). A wireless sensor network infrastructure for personal monitoring. In *Proceedings of the 5th International Workshop on Wearable Micro- and Nanosystems for Personalised Health*, Valencia, Spain.

Barlow, J., Wright, C., Sheasby, J., Turner, A., & Hainsworth, J. (2002). Self management approaches for people with chronic conditions: A review. *Patient Education and Counseling*, *48*, 177–187. doi:10.1016/S0738-3991(02)00032-0

Barry, C. A., Bradley, C. P., Britten, N., Stevenson, F. A., & Barber, N. (2000). Patients' unvoiced agendas in general practice consultations: Qualitative study. *British Medical Journal*, 1246–1250. doi:10.1136/bmj.320.7244.1246

Bodenheimer, T., Lorig, K., Holman, H., & Grumbach, K. (2002). Patient self management of chronic disease in primary care. *Journal of the American Medical Association*, *288*(19), 2469–2475. doi:10.1001/jama.288.19.2469

Borrel, M. (1993). Training the senses, training the mind . In Bynum, W. F., & Porter, R. (Eds.), *Medicine and the five senses* (pp. 244–261). Cambridge, UK: Cambridge University Press.

Buchholz, T., Krause, M., Linnhoff-Popien, C., & Schiffers, M. (2004) CoCo: Dynamic composition of context information. In *Proceedings of the First Annual International Conference on Mobile and Ubiquitous Computing*, Boston, MA (pp. 335-343).

Cahill, J. (1996). Patient participation: A concept analysis. *Journal of Advanced Nursing*, *24*, 561–571. doi:10.1046/j.1365-2648.1996.22517.x

Callaway, E., & Callaway, E. Jr. (2003). *Wireless sensor networks: Architectures and protocols*. Boca Raton, FL: CRC Press.

Conrad, P. (1985). The meaning of medications: Another look at compliance. *Social Science & Medicine*, 29–37. doi:10.1016/0277-9536(85)90308-9

Continua Health Alliance. (2010). *Your health. Connected.* Retrieved from http://www.continuaalliance.org

Cranor, L. F. (1999). *Beyond concern: Understanding net users' attitudes about online privacy*. New York, NY: CORR Publications.

Crespo, R. (2007). Virtual community health promotion. *Preventing Chronic Disease*, *4*(3), 75.

Davy, S., Barrett, K., Serrano, M., Strassner, J., Jennings, B., & van der Meer, S. (2007). Policy interactions and management of traffic engineering services based on ontologies. In *Proceedings of the Latin American Network Operations and Management Symposium.*

Della Mea, V. (2001). What is e-Health (2): The death of telemedicine? *Journal of Medical Internet Research*, *3*(2), 22. doi:10.2196/jmir.3.2.e22

Elmisery, A. M., & Fu, H. (2010). Privacy preserving distributed learning clustering of healthcare data using cryptography protocols. In *Proceedings of the 34th IEEE Annual International Computer Software and Applications*, Seoul, Korea.

European Commission. (2009). *Accelerating the development of the ehealth market in Europe.* Retrieved from http://ec.europa.eu/information_society/activities/health/docs/publications/lmi-report-final-2007dec.pdf

Ferguson, T. (2010). *ePatients: How they can help us heal health care*. Retrieved from http://www.e-patients.net/e-Patients_White_Paper.pdf

Gatzoulis, L., & Iakovidis, I. (2008, May 21-23). The evolution of personal health systems. In *Proceedings of the 5th International Workshop on Wearable Micro- and Nanosystems for Personalised Health*, Valencia, Spain.

Giustini, D. (2006). How Web 2.0 is changing medicine. *British Medical Journal*, 1283–1284. doi:10.1136/bmj.39062.555405.80

Gray, P., & Salber, D. (2001). Modelling and using sensed context information in the design of interactive applications. In M. R. Little & L. Nigay (Eds.), *Proceedings of the 8th IFIP Working Conference on Engineering for Human-Computer Interaction*, Toronto, ON, Canada (LNCS 2254, pp. 317-335).

HEALTHSERVICE24. (2010). *Mobile healthcare solution - a life less limited.* Retrieved from http://www.healthservice24.com

Herzog, R. (2006, January). Mobile patient monitoring – applications and value propositions for personal health. In *Proceedings of the International Workshop on Wearable Micro- and Nanosystems for Personalised Health*, Luzern, Switzerland.

Hochstatter, I., Duergner, M., & Krause, M. (2007). A context middleware using an ontology–based information model. In A. Pras & M. van Sinderen (Eds.), *Proceedings of the 13th Open European Summer School and IFIP TC6.6 Workshop on Dependable and Adaptable Networks and Services* (LNCS 4606, pp. 17-24).

International Telecommunication Union (ITU). (2008). *Implementing e-health in developing countries: Guidance and principles*. Retrieved from http://www.itu.int/ITU-D/cyb/app/docs/e-Health_prefinal_15092008.PDF

Judd, G., & Steenkiste, P. (2003). Providing contextual information to pervasive computing applications. In *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications*, Seattle, WA (pp. 133-142).

Levenstein, J. H., McCracken, E. C., McWhinney, I., Steward, M. A., & Brown, J. B. (1986). The patient centred clinical method: A model for the doctor-patient interaction in family medicine. *Family Practice*, *3*, 24–30. doi:10.1093/fampra/3.1.24

Lupu, E., Dulay, N., Sloman, M., Sventek, J., Heeps, S., & Strowes, S. (2008). AMUSE: Autonomic management of ubiquitous e-health systems. *Concurrency and Computation*, *20*(3), 277–296. doi:10.1002/cpe.1194

Mantzana, V., & Themistocleus, M. (2004). Identifying and classifying benefits of integrated healthcare systems using an actor oriented approach. *Journal of Computing and Information Technology*, 265-278.

MOBILHEALTH. (2010). *Innovative GPRS/UMTS mobile services for applications in healthcare.* Retrieved from http://www.mobihealth.org

Mol, A. (2008). *The logic of care and the problem of patient choice*. London, UK: Routledge.

Montón, E., Hernández, J. F., Blasco, J. M., Hervé, T., Micallef, J., & Grech, I. (2008). A body area network for patient wireless monitoring. *IEEE Communications*, *2*(2), 215–222.

Neves, P. A. C. S., Ferreira, D. J. M., Esteves, D., Felix, D. R. M., & Rodrigues, J. J. P. C. (2008, September 25-27). InHand – mobile professional context and location aware tool. In *Proceedings of the International Conference on Software, Telecommunications and Computer Networks*, Dubrovnik, Croatia (pp. 80-84).

Neves, P. A. C. S., Fonseca, J. F. P., & Rodrigues, J. J. P. C. (2008, January 28-31). Simulation tools for wireless sensor networks in medicine: A comparative study. In *Proceedings of the BIOSTEC International Joint Conference on Biomedical Engineering Systems and Technologies*, Madeira, Portugal.

Opie, A. (1998). Nobody asked me for my view: Users' empowerment by 50 multidisciplinary health teams. *Qualitative Health Research*, *8*(2), 188–206. doi:10.1177/104973239800800204

Preece, J. (1998). Emphatic communities: Reaching out across the web. *Interaction*, *5*(2), 32–43. doi:10.1145/274430.274435

Sandars, J., & Haythornthwaite, C. (2007). New horizons for e-learning in medical education: Ecological and Web 2.0 perspectives. *Medizinische Technik*, *29*(4), 307–310.

Serrano, M. Ó'Foghlú, M., Fernstrom, M., & Storni, C. (2009). *Future perspectives under the HEA-FutureComm serving society project.* Paper presented at the Future Internet Assembly, Stockholm, Sweden.

Serrano, M. Ó'Foghlú, M., Storni, C., & Fernström, M. (2009, June 24-26). *Autonomic computing systems acting as patient self-management support in early and quick medical assistance.* Paper presented at the 6th International Workshop on Wearable Micro- and Nanosystems for Personalised Health, Oslo, Norway.

Serrano, M. Ó'Foghlú, M., & Donnelly, W. (2010, May 26-28). Patient monitoring and autonomic systems as integral support in early cardio-vascular diagnostics. In *Proceedings of the 6th International Workshop on Wearable Micro- and Nanosystems for Personalised Health*, Berlin, Germany.

Serrano, M., Serrat, J., Strassner, J., & Foghlú, Ó, M. (2007). Management and context integration based on ontologies, behind the interoperability in autonomic communications. *System and Information Sciences Notes*, *1*(4), 435–442.

Storni, C. (2010). Multiple forms of appropriation in self-monitoring technology. *International Journal of Human-Computer Interaction*, *26*(5), 553–561. doi:10.1080/10447311003720001

Trummer, U. F., Mueller, U. O., Nowak, P., Stidl, T., & Pelikan, J. M. (2006). Does physician–patient communication that aims at empowering patients improve clinical outcome? A case study. *Patient Education and Counseling*, *61*(2), 299–306. doi:10.1016/j.pec.2005.04.009

Wagner, E. H., Glasgow, R. E., Davis, C., Bonomi, A. E., Provost, L., & McCulloch, D. (2001). Quality improvement in chronic illness care: A collaborative approach. *Joint Commission Journal on Quality and Patient Safety*, *27*(2), 63–80.

Zhu, Y., Keoh, S. L., Sloman, M., & Lupu, E. C. (2009). A lightweight policy system for body sensor networks. *IEEE Transactions Network and Service Management*, *6*(3), 137–148. doi:10.1109/TNSM.2009.03.090301

Martín Serrano received PhD in Theory Signal and Communications (Hons) with special "European Doctor" mention (CumLaude) and M.Sc. European Master in Research and Information Technologies-MERIT both from Universitat Politécnica de Catalunya (UPC), Spain. He graduated as Communications and Electronic Engineer (Hons) from School of Mechanical and Electrical Engineering (ESIME-UC). He received B.Sc. in Electronics from CECyT 11, both from Instituto Politécnico Nacional (IPN), México. He joined the Telecommunications Software and Systems Group (TSSG) at the Waterford Institute of Technology (WIT), Ireland as Research Fellow in 2008. He has extensive research background actively participating in EU-IST projects and EU-Research Networks of Excellence since 2002 and industry experience working for National Panasonic/AKME-BC as team supervisor on Engineering Research, Planning, Management and development of new technologies (1998-2002). His research activity includes Pervasive Computing, Knowledge Engineering, Autonomic Communications. He is also investigating sensor networks to monitor health data patterns for personalized healthcare systems (pHealth).

Appendix E: Pervasive Healthcare Service Health System Scenario

Article XV

Diagnosis Support on cardio-Vascular Signal monitoring by using Cluster Computing

Ahmed M. Elmisery, Martín Serrano, Dmitri Botvich

In Proceedings of the 3rd KES International Conference on Intelligent Decision Technologies (IDT 2011), Piraeus, Greece, July 2011.

Diagnosis Support on Cardio-Vascular Signal Monitoring by using Cluster Computing

Ahmed M. Elmisery, Martín Serrano and Dmitri Botvich

Telecommunications Software & Systems Group, Waterford Institute of Technology, Waterford, Ireland.

Abstract. The support for remote data processing and analysis is a necessary requirement in future healthcare system. Likewise interconnect/manage medical devices and distributed processing of data collected through these devices are crucial processes for supporting personalised healthcare systems. This work introduces our research efforts to build a monitoring application hosted on a cluster computing environment supporting personalised healthcare systems (pHealth). The application is based on a novel distributed clustering algorithm that is used for medical diagnosis of cardio-vascular signals. The algorithm collects different statistics from the cardiac signals and uses these statistics to build a distributed clustering model automatically. The resulting model can be used for diagnosis purposes of cardiac signals. A cardio-vascular monitoring scenario in cluster computing environment is presented and experimental results are described to demonstrate the accuracy of cardio-vascular signals diagnosis. Advantages of using data analysis techniques and cluster computing in medical diagnosis also discussed in this work.

Keywords: Personalised Health Systems, ICT enabled Personal Health, Health Monitoring, Pervasive Computing on eHealth.

1 Introduction

Trends in the next generation of healthcare systems demand applications that can allow prevention of diseases even before they are apparent by using assisted sensors and networks (Yanmin et al. 2009; Lupu et al. 2008). Personalised healthcare systems (pHealth) (Gatzoulis,Iakovidis 2008) is one application that can achieve this objective by presenting a personalized healthcare services. With personalised healthcare services people can receive more accurate diagnostics and early medical assistance. Designing these systems according to individual requirements and based on health data being collected from wearable sensors is challenging task. These systems demands a local processing for the health data and capabilities for distributed data analysis (Herzog et al. 2006) as well as a net-

work infrastructure with high performance (ICT's) to be able to react in real-time to variations in the data. Cluster Computing and other distributed computing environments have demonstrated their advantages in pHealth systems by offering scalability, availability as well as ability to process massive amount of data (Neves et al. 2008). However, privacy of health data is a main requirement that must be taken into consideration when developing pHealth systems in these environments.

Modern medicine can benefits from pHealth systems by building user's health profiles that can offer personalised support, early assistance, accurate diagnostics and quick response when symptomatic diseases are detected during the local and remote analysis of these profiles. Also, pHealth systems provide procedures to support monitoring the progress of diseases as well as their therapeutic intervention. A key goal in pHealth systems is the ability to perform analysis on either data taken during normal activities or data based on regular medical checks. As a consequence, the people activity/freedom is not affected and accurate results can be attained. Modern pHealth systems allow people to continue their activities and envisage a real time and interactive environment for patient-doctor information exchange. A clear advantage when using these systems is to offer accurate diagnostics for remote healthcare subscribers.

We concentrated on distributed clustering as an analysis tool to support healthcare services. This work presents our efforts to build a framework for personalised healthcare applications management. The main objective for this research is to introduce an application for distributed learning clustering (DLC) algorithm (Elmisery,Huaiguo 2010) in the diagnosis of cardiovascular signals. The rest of the work is organized as follow: Section 2 discusses cluster computing as a processing environment to support personalised healthcare applications. Section 3 describes research work results as part of integral cardio-vascular monitoring system in the framework for personalised healthcare applications management introduced in this work. And finally Section 4 summarizes the research advances and concludes this work.

2 Cluster Computing Environment Supporting Personalised Healthcare Applications

This research introduces a framework for personalised healthcare applications management that can manage different healthcare applications running in the same computing environment. This framework is hosted in a cluster computing environment to support massive health data analysis, distributed data storage and health communication networks see Figure (1). Cluster computing play an important role as a processing environment for health data; as it empowers the execution of different health application and the exchange of the data between them.

The end-user (i.e. patient or healthy people) has a main role to supply the applications' databases with his/her health data. This allows these applications to

build accurate models for diagnosis and monitoring of health status. Also, the end-user has an important role in the evaluation and enhancement of these applications.

The development of user centred systems is crucial and highlights the end-user role in healthcare research and technological development practices. Personalised healthcare applications require an active role for the end-user, as he/she submits health data to the health applications then he/she implies the correct understanding of the medical information provided by the health application. This feature acts as a playground for the healthcare applications to develop a new applications and services.

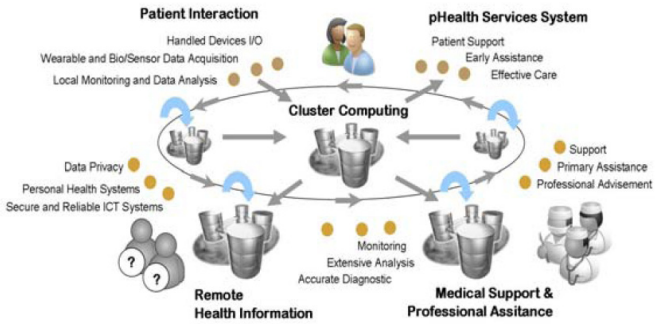


Fig. 1. Cluster computing as to support personalised healthcare applications

We assume that patients keen to build a local knowledge in order to deal with the alternative solutions of their health problems. The information obtained by end-users can help to enrich health knowledge and research activities. For example if a drug is being consumed by mid-long term period of time. It is difficult and expensive to track the side-effects for it in order to improve or change that drug, but if the patients play the role of self monitoring assisted with ICT's, they can provide valuable data to assist medical professionals in this task.

3 Personalized Medical Support for Cardio-Vascular Monitoring

This section describes related interdisciplinary application for cardio-vascular monitoring in the framework of personalised healthcare systems. In this application, we employ data clustering techniques to group different cardiovascular signals in order to assign a patient to a physiological condition using no prior knowledge about disease states. We used a new clustering algorithm called distributed learning clustering algorithm (DLC). DLC is based on the idea of stage clustering and offers many advantages than current clustering algorithms, as following:

- The algorithm produces clusters with acceptable accuracy; these clusters have different shapes, sizes and densities.
- The algorithm was designed with the goal of enabling a privacy preserving version of the data.
- The algorithm helps the user to select proper values for its parameters, and tune parameters for better results.
- The algorithm present different statistics for clustering validity in each stage, and use these statistics to enhance the resulting clusters automatically.
- The applicability in the algorithm to work in networked environments (p2p, cluster computing or grid systems).

Figure (2) depicts the different processes inside our proposed personalized medical application that is used for supporting the diagnosis of cardiovascular signals. In order to enhance the model building process in that application, we proposed an adaptive strategy that utilizes both patient cardiovascular signals and established ECG medical databases, that is more suitable for remote diagnosis. The process described as following:

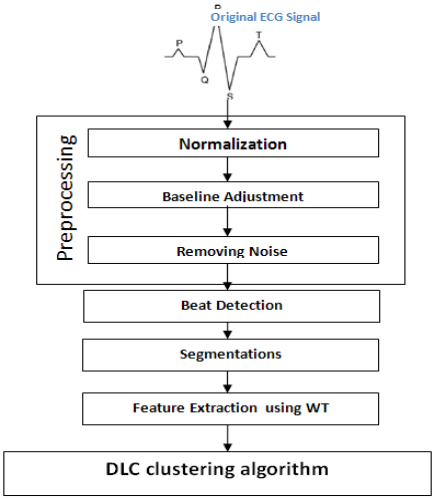


Fig. 2. Personalized Medical Application for Early Cardio-vascular Diagnostics

1. Use the MIT BIH Arrhythmia database (Moody,Mark 1990) to build an initial clustering model.
2. Test the model on the patient.
3. Collect the new ECG data from this patient.
4. Store the records that achieve high error values beyond a predefined threshold in different Database.
5. Send these data to cardiologists for detailed analysis. This process is done off-line.

6. Collect the cardiologists' annotation and use these data in the model tuning process.

ECG recordings carry significant information about the overfull behavior of cardiovascular system and physiological patient conditions. The ECG signal is pre-processed to remove noise and abnormal features, extract features and select certain features that will have high influence on our DLC clustering algorithm. The relevant information is encoded in the form of feature vector that is used as input for DLC algorithm. The key goal for the DLC algorithm is to be able to find patterns in the ECG signals that effectively discriminate between different conditions or categories under investigation.

3.1 ECG Signal Analysis

This section introduces the formalism used for data analysis (Clifford et al. 2006). In the start, each signal is pre-processed by normalization process which is necessary to standardize all the features to the same level. After that, we adjust the baseline of the ECG signal at zero line by subtracting the median of the ECG signal (Yoon et al. 2008). ECG signals can be contaminated with several types of noise, so we need to filter the signal to remove the unwanted noise. ECG signals can be filtered using Low pass filter, high pass filter and Notch filter (Chavan et al. 2008). As shown in figure (3), the ECG signal consists of P-wave, PR-interval, PR-segment, QRS complex, ST-segment, and T-wave. The QRS complex is very important signal that is useful in the diagnosis of arrhythmias diseases. In general the normal ECG rhythm means that there is a regular rhythm and waveform. Correct detection of QRS-complexes forms the basis for most of the algorithms used in automated processing and analysis of ECG (Kors,Herpen 2001).

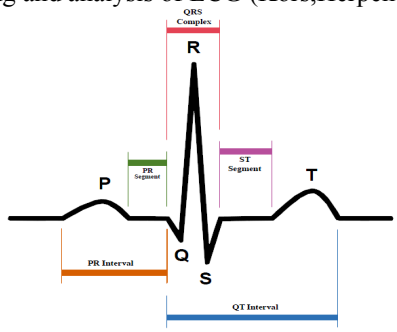


Fig. 3. ECG Signal Analysis Process Using QRS Metrics (Atkielski 2006)

However, the ECG rhythm in a patient with arrhythmia will not be regular in certain QRS complex (Dean 2006). Our QRS detection algorithm must be able to detect a large number of different QRS morphologies in order to be clinically useful and able to follow sudden or gradual changes of the prevailing QRS morphol-

ogy. Also it should help to avoid errors related to false positives due either to artifacts or high amplitude T waves. On the other side, false negatives may occur due to low amplitude R waves.

3.2 Clustering Analysis for ECG Signal

Clustering analysis aims to group collection of signals or cases into meaningful clusters without need to prior information about the classification of patterns. There is no general agreement about the best clustering algorithm (Xu,Wunsch 2005); different algorithms reveal certain aspects of the data based on the objective function used. The clustering algorithm learns by discovering relevant similarity relationships between patterns. The result of applying such algorithms is groups of signals evince recurrent QRS complexes and /or novel ST Segments; where each group can be linked to significant disease or risk.

Detecting relevant relationships between signals addressed in the literature using different clustering algorithms. For example, the work in (Iverson et al. 2005) applied point wise correlation dimension to analysis of ECG signals from patients suffer from depression. The results obtained in this study indicate that clustering analysis able to discriminate clinically meaningful clusters with and without depression based on ECG information. Authors in (Dickhaus et al. 2001; Bakardjian 1992) cluster collected ECG data into clinically relevant groups without any prior knowledge. This emphasized the advantage of clustering in different classification problems especially in exploratory data analysis or when the distribution of the data is unknown.

For detecting the R-peaks in the ECG signal (y[k]), we use an algorithm proposed in(S. et al. 1997). It starts searching for local modulus maxima at large scale then at fine ones. This procedure reduces the effect of high frequency noise; also it uses adaptive time amplitude threshold and refractory period information and rejects isolated and redundant maximum lines (artifacts, high amplitude T wave or low amplitude R waves). Detecting R-peaks starts with calculating zero crossing of the wavelet between a positive maximum- negative minimum that is marked as R-peak (m_{zc}). Once R-peaks are found, the RR-interval between each two consecutive heartbeats is computed by:

$$RR(e) = m_{zc}(e+1) - m_{zc}(e) \quad (1)$$

Where e refers to heartbeat sequence index. For heartbeat segmentation purposes, starting and ending points are obtained as follows:

$$y[k] = y[m_{zc}(e) - 0.25RR(e) : m_{zc}(e) + 0.75RR(e)] \quad (2)$$

The length of this interval is different for each heartbeat; figure (4) illustrate the detection of RR-interval. The length variability is removed by means of trace segmentation.

Following that, Feature extraction is performed using WT decomposition. The heartbeats will represented as an array of time-varying duration, In order to com-

pare the heartbeat morphologies it is necessary to use a proper dissimilarity measure for DLC algorithm. In this work, we used dynamic time warping (DTW) used in (Cuesta-Frau et al. 2007) to find an optimal alignment function between two sequences of different length. The heartbeat is considered if its dissimilarity measure with other elements in the resulting set is higher than a specific threshold. The DLC clustering can be expressed as following:

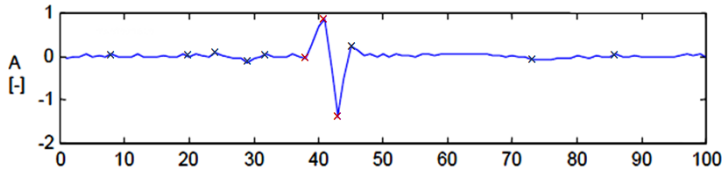


Fig. 4. Illustration for the detection of RR-interval in ECG Signal

Consider H is the set of n heartbeats, the goal of local learning and analysis LLA step is to find $R \in H$, with i beats, where $i < n$. All dissimilar heartbeat is represented in $R = s_1, \ldots . s_i$ and similar ones are omitted. Then in distributed clustering step (DC) step the set $R = s_1, \ldots . s_i$ is partitioned to a set of clusters $C = c_1, \ldots . c_m$,where each cluster contains proportionate heartbeats. Table 2, shows the resulting heartbeats after the execution of LLA step.

Set of heartbeats used in experiment							
label	Normal	Lbbb	Rbbb	PVC	Ap	P	Total
No.beats	9870	7361	6143	8450	2431	7340	41595

Table 1. Heartbeat used

Resulting heartbeats after Pre-processing and LLA							
label	Normal	Lbbb	Rbbb	PVC	Ap	P	Total
No.beats	1730	1320	861	1763	935	843	7452

Table 2. Resulting heartbeats after LLA

Label	Meaning
Normal	Normal beat
Lbbb	Left bundle branch block beat
Rbbb	Right bundle branch block beat
PVC	Premature ventricular contraction
Ap	Atrial premature beat
P	Paced rhythm

Table 3. Abbreviations Used

Our first experiment done on DLC to measure its accuracy in determining different heartbeat clusters, The figure (5) shows the relation between merge error in

DC stage and the number of clusters. As shown in figure (5), the merge error (LET) decreases which indicates only equivalent heartbeat clusters are being merged.

In order to evaluate the performance of our algorithm, we used two error metrics defined in (Cuesta-Frau et al. 2003) . The first metric is clustering error (CR) which is the percentage of heartbeats in a cluster that do not correspond to the class of such cluster. Second metric is the critical error (CIE) which is the number of heartbeats in a class that do not have a cluster and are therefore included in other's classes' clusters.

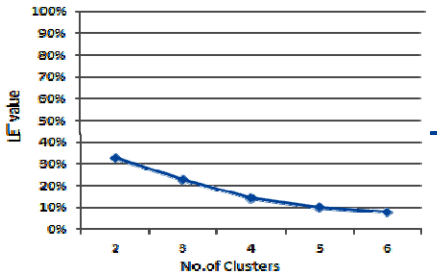
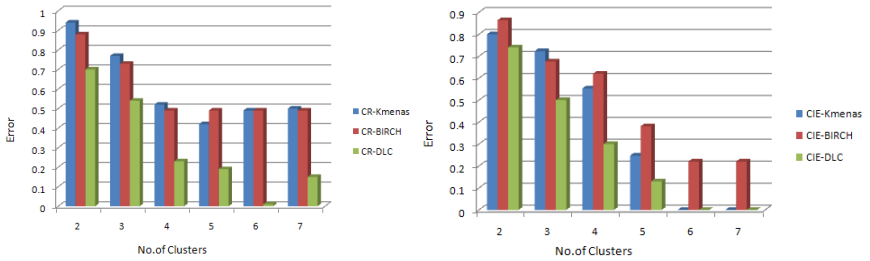


Fig. 5. Relation between Different Clusters and Merge Error



(a) (b)

Fig. 6. (a) The Values of CR for Different No. of Clusters. (b) The Values of CIE for Different No. of Clusters

In the second experiment, we want measure the relation between different no. of clusters and the values of clustering error (CR) and critical error (CIE). Based on figure 6(a) and (b), we can deduce that both CR and CIE for DLC algorithm decrease with the increase in no. of clusters till reaching correct number of clusters. In The third experiment, we compare the results of DLC with other clustering algorithms, here we select BIRCH and k-means; we tune the parameters in each algorithm to get the same number of clusters. Figure 6(a) and (b) contain both CR and CIE values for each algorithm for different number of clusters. The results show the accuracy of the results achieved using DLC compared to other algorithms.

3.3 Privacy in Clustering Cardiovascular Data

Privacy aware users consider ECG signals sensitive information, as these signals allow the health application providers to infer different mental condition for the patients (depressed, afraid, walking or running… etc). As a consequence, they require certain levels of privacy and anonymity in handling their signals. Our aim is to permit clustering of ECG signals without learning any private information about the patient. In reality, these signals do not need to be fully disclosed to the healthcare provider in order to build an accurate model. We preprocess the wavelets coefficients using LLA step to build up sets of initial clusters where the end-user patters are compared with each other locally, then we take the representatives of each initial cluster as an input to the distributed clustering (DC) step. These representatives used as pattern reference to associate clusters patters with same diseases. Also LLA uses wavelets transformation to preserve privacy for ECG signals by decomposing wavelet coefficients .These two steps affect on both accuracy of the results and privacy level attained.

4 Conclusions

This work has introduced our vision for a personalized health systems based on monitoring ECG signals as an application example. Research efforts have been conducted to promote cluster algorithm as an alternative solution for finding out data similarities between cardio-vascular patterns and clusters previously diagnosed/detected. We have introduced a novel solution using DLC algorithms to cluster morphological similar ECG signals and enforcing privacy when matching these patterns. Experimental results were done in set of ECG recordings from MIT database. DLC yielded 99.9% clustering accuracy considering pathological versus normal heartbeats. Both clustering error and critical error percentage was 1%. We will continue investigating computing techniques to map cardiac patterns for different heart diseases and produce reactive solutions in the communications systems.

References

Atkielski, A.: Electrocardiography. In. Wikipedia, (2006)

Bakardjian, H.: Ventricular beat classifier using fractal number clustering. Medical and Biological Engineering and Computing **30**(5), 495-502 (1992). doi:10.1007/bf02457828

Chavan, M.S., Agarwala, R.A., Uplane, M.D.: Interference reduction in ECG using digital FIR filters based on rectangular window. WSEAS Trans. Sig. Proc. **4**(5), 340-349 (2008)

Clifford, G.D., Azuaje, F., McSharry, P.: Advanced Methods And Tools for ECG Data Analysis. (2006)

Cuesta-Frau, D., Biagetti, M., Quinteiro, R., Micó-Tormos, P., Aboy, M.: Unsupervised classification of ventricular extrasystoles using bounded clustering algorithms and morphology matching. Medical and Biological Engineering and Computing **45**(3), 229-239 (2007). doi:10.1007/s11517-006-0118-1

Cuesta-Frau, D., Pérez-Cortés, J.C., Andreu-García, G.: Clustering of electrocardiograph signals in computer-aided Holter analysis. Computer Methods and Programs in Biomedicine **72**, 179-196 (2003). doi:10.1016/s0169-2607(02)00145-1

Dean, G.: How Web 2.0 is changing medicine, vol. 333. vol. 7582. British Medical Association, London, ROYAUME-UNI (2006)

Dickhaus, H., Maier, C., Bauch, M.: Heart rate variability analysis for patients with obstructive sleep apnea. In: Engineering in Medicine and Biology Society, 2001. Proceedings of the 23rd Annual International Conference of the IEEE, 2001 2001, pp. 507-510 vol.501

Elmisery, A.M., Huaiguo, F.: Privacy Preserving Distributed Learning Clustering of HealthCare Data Using Cryptography Protocols. In: Computer Software and Applications Conference Workshops (COMPSACW), 2010 IEEE 34th Annual, 19-23 July 2010 2010, pp. 140-145

Gatzoulis, L., Iakovidis, I.: The Evolution of Personal Health Systems. Paper presented at the 5th pHealth Workshop on Wearable Micro and Nanosystems for Personalised Health, Valencia-Spain,

Herzog, R., Konstantas, D., Bults, R., Halteren, A.V., Wac, K., Jones, V., Widya, I., Streimelweger, B.: Mobile Patient Monitoring - applications and value propositions for personal health. Paper presented at the pHealth 2006, the International Workshop on wearable micro- and nanosystems for personalized Health, Luzern, Switzerland,

Iverson, G., Gaetz, M., Rzempoluck, E., McLean, P., Linden, W., Remick, R.: A New Potential Marker for Abnormal Cardiac Physiology in Depression. Journal of Behavioral Medicine **28**(6), 507-511 (2005). doi:10.1007/s10865-005-9022-7

Kors, J.A., Herpen, G.v.: The Coming of Age of Computerized ECG Processing: Can it Replace the Cardiologist in Epidemiological Studies and Clinical Trials? In., pp. 1161-1165. (2001)

Lupu, E., Dulay, N., Sloman, M., Sventek, J., Heeps, S., Strowes, S., Twidle, K., Keoh, S.-L., Schaeffer-Filho, A.: AMUSE: autonomic management of ubiquitous e-Health systems. Concurr. Comput. : Pract. Exper. **20**(3), 277-295 (2008). doi:10.1002/cpe.v20:3

Moody, G.B., Mark, R.G.: The MIT-BIH Arrhythmia Database on CD-ROM and software for use with it. In: Computers in Cardiology 1990, Proceedings., 23-26 Sep 1990 1990, pp. 185-188

Neves, P.A.C.S., Fonsec, J.F.P., Rodrigue, J.J.P.C.: Simulation Tools for Wireless Sensor Networks in Medicine: a Comparative Study. Paper presented at the International Joint Conference on Biomedical Engineering Systems and Technologies, Funchal, Madeira-Portugal,

S., S.J., N., T.S., P., B.R.K.: USING WAVELET TRANSFORMS FOR ECG CHARACTERIZATION : AN ON-LINE DIGITAL SIGNAL PROCESSING SYSTEM, vol. 16. vol. 1. Institute of Electrical and Electronics Engineers, New York, NY, ETATS-UNIS (1997)

Xu, R., Wunsch, D.: Survey of Clustering Algorithms. Neural Networks, IEEE Transactions on **16**(3), 645-678 (2005). doi:citeulike-article-id:469342

Yanmin, Z., Sye Loong, K., Sloman, M., Lupu, E.C.: A lightweight policy system for body sensor networks. Network and Service Management, IEEE Transactions on **6**(3), 137-148 (2009)

Yoon, S.W., Min, S.D., Yun, Y.H., Lee, S., Lee, M.: Adaptive Motion Artifacts Reduction Using 3-axis Accelerometer in E-textile ECG Measurement System. J. Med. Syst. **32**(2), 101-106 (2008). doi:10.1007/s10916-007-9112-x

Appendix E: Pervasive Healthcare Service Health System Scenario

Article XVI

A Distributed Collaborative Platform for Personal Health Profiles in Patient-Driven Health Social Network

Ahmed M. Elmisery, Dmitri Botvich

Submitted to the the Taylor and Francis International Journal of New Review of Hypermedia and Multimedia.

A Distributed Collaborative Platform for Personal Health Profiles in Patient-Driven Health Social Network

Ahmed M. Elmisery and Dmitri Botvich

[1]TSSG, Waterford Institute of Technology, Co. Waterford, Ireland

ahmedmohmed2001@gmail.com, dbotvich@tssg.org

Abstract. With the rapid growth of social media platforms, more purpose driven social networking sites have emerged and gained much popularity. In healthcare, Health Social Networks (HSNs) have become an integral part of healthcare to augment the ability of people to communicate, collaborate and share information in the health care domain despite obstacles of geography and time. Doctors disseminate relevant medical updates in these platforms and Patients take into account opinions of strangers when making medical decisions. This paper introduces our efforts to develop a core platform called Distributed Platform for Health Profiles (DPHP) that enables individuals or groups to control their personal health profiles. DPHP stores user's personal health profiles in a non-proprietary manner which will enable healthcare providers and pharmaceutical companies to reuse these profiles in parallel in order to maximize the effort where users benefit from each usage for their personal health profiles. DPHP also facilitates the selection of appropriate data aggregators and assessing their offered datasets in an autonomous way. In addition, DPHP preserves the privacy of the merged health profiles from multiple sources, which were involved within the data aggregation. Experimental results were described to demonstrate the proposed search model in DPHP platform. Multiple advantages might arise when healthcare providers utilize DPHP to collect data for various data analysis techniques in order to improve the clinical diagnosis and the efficiency measurement for some medications in treating certain diseases. In addition to other related healthcare areas where were discussed in this paper.

Keywords: Personal Heath Profiles; Community; Health Social Networks; ICT enabled Personal Health

1 Introduction

The raise of social networks as an effective tool for the interaction between people and as a platform for sharing their health conditions, leads to the appearance of more purpose driven social networks in healthcare. Utilizing social networks as an integral part of healthcare has made a significant impact in digital healthcare and the emerging of what is referred to as Health social networks (HSNs). Health Social networks hold a considerable potential value for health care organizations [1] because they fetch people together for collaboration and collect information related to their experiences and reflections. One-third of Americans who go online try to find fellow patients similar to their health status to discuss their conditions [2] and 36% of the users utilize other users' information and opinions on social networks before making medical decisions[3]. Health Social networks (HSNs) [1] were initially directed at patients but different caretakers and researchers may be able to participate within it. HSNs hold a considerable potential value for health care organizations because they can be used to reach collaborators, accumulate information and facilitate an effective partnership. However, Trends in the next generation of healthcare systems demand applications that can allow prevention of diseases even before they are apparent by using advanced analytics and learning techniques [4, 5].

Health social networks can also be employed to provide real dataset regarding clinical trials. The existence of health social networks makes traditional clinical trials more efficient through the availability of large searchable online databases of patients' information which contains their health history and conditions. Pharmaceutical firms, healthcare analysts, health policy planners and other interested parties can assess the demand and market size directly from health social network websites. To date, there are numerous paradigms for health social networks that exist on the internet including PatientsLikeMe®, DailyStrength®, CureTogether®, peoplejam® and OrganizedWisdom®. The largest and well-known health social network is PatientsLikeMe which launched in 2004, and it hits a new milestone of 100,000 members as of June 2011. PatientsLikeMe® and Inspire® are an example of two health social networks offering access to clinical trials, selling anonymized data to pharmaceutical companies, universities and medical research labs. As an example for low cost patient recruitment using HSNs, in May 2008, Novartis recruited clinical trial participants from PatientsLikeMe® estimating that they could reduce the time required for their study of a new medicine for only a few months [6]. In another case, PatientsLikeMe® was utilized to gather ALS patients for a research project and this project has managed to collect 50 DNA samples [7]. This effect might not seem high but the time and cost savings in recognizing, inspecting, contacting and obtaining responses from relevant patients is critical.

HSNs can lead to discovering new findings that can help to understand natural history and development of various diseases by utilizing quantitative analysis tools

on massive data that is gathered through various patients' communities who are continuously interacting and reporting their health conditions and medical history. For example, PatientsLikeMe® has an in-house research staff which is publishing some of their healthcare research, such as their research that is related to determining the non-motor symptoms of Parkinson's disease in younger patients [8]. HSNs are equipped with health tracking process that can be employed by patients to provide their experience and feedback to the clinical trials process including their response to the drugs. For example, patients registered in PatientsLikeMe® network have noticed and suggested a set of corrections and improvements to the graphical display of the data in ALS clinical trials [9].

The next generation of HSNs is based on patient-inspired research, which is also called crowd-sourced health research. These novel HSNs were emerged as experienced patients may no longer have the willingness to wait for formal research findings and medical clinical trials, and can possibly fill the gap for rare diseases that do not make outstanding business cases in the existing Healthcare model. The experienced patients can study and review research literature on their own and investigate new findings, tracking the results, sharing the information and running non-traditional clinical trials with themselves. As an example, a patient registered in PatientsLikeMe®, diagnosed with rapidly progressive and young-onset ALS, managed to collected information regarding other 250 patients regarding a self-experiment with lithium [10] for a research study. This patient-inspired research had found [11] preliminary results regarding the use of lithium as a therapy does not slow the disease progression. This example highlights the power of patient-inspired research and role of patients in medical research. The ownership of that healthcare process and the concomitant controversial legal, ethical, methodological is other issues. However, fraud and privacy breaches are likely to arise in HSNs as there are significant economic incentives for drugs and other treatments to have high patient usage statistics and favorable reputations. This requires a platform that is able to select data in a more rational and similar way to human ones only in a shorter period of time autonomously and automatically while preserving the privacy of participants.

This paper introduces a proposed platform that we called "Distributed Platform for Health Profiles" (DPHP) that can extract helpful datasets for clinical trials and detect fraudulent aggregators. DPHP utilizes a search model that considers multiple attributes of various data aggregators and their offered data, such as success criterion and trust rank for each aggregator beside price, type, accuracy level, anonymization level, tuples types, no of records, gathering method and demographic for each dataset offered by such aggregator. Furthermore, DPHP facilitates a tendering process where aggregators tender their personal health data in an intelligent manner. Privacy concerns for the participants have obliged DPHP to utilize the privacy enhancing framework proposed in [12-17] in order to give the patients confidence that the usage and disclosure of their healthcare profiles and related demographic information are under their control. This work is structured as follows. In Section 2, related works are described. Section 3, briefly introduces the proposed DPHP platform (Distributed Platform for Health Profiles). Section 4, describes our proposed fuzzy search model and Section 5 presents a case study to illustrate this fuzzy search model on proposed platform. Section 6 concludes this paper.

2 Related Works

The current literature addresses the problem of exploiting social data from the prospective of knowledge sharing. In some systems, very general techniques like the ones that were exploited in the information filtering research are used to search the heterogeneous information sources with little information available about the users' needs. The users should be assisted while exploring data in social data, the system should keep track of their actions to identify their real needs in order to extract suitable data that is matching their needs. In [18, 19] a peer to peer approach is proposed based on the users' communities concept, where the community will have an aggregate user profile representing the group as a whole but not the individual users. Communication occurs between the individual users but not with the servers. Thus, the processing is done at the client side. Storing users' profiles on their own side and running the required processing in a distributed manner without relying on any server is another approach proposed in [20]. While those techniques are suited in dealing with large scale applications, other works have shown the need for more purpose specific techniques to be applied in order to personalize the search process on the social data. The work in [21] describes a recommender system for VOD applications, where the structure of a movie database is exploited to customize the recommended items for the users. The system analyzes customers' selections in order to identify the items' attributes which are affecting their decisions. This information aids in filtering out the new items in order to select the items to be recommended. The work in [22] presented a system to generate labels for museum items by summarizing the information stored in the records of an external database. This information consists of unstructured natural language text, where the system exploits NLP techniques to interpret the text and then generates summaries based on the detailed domain ontology. This deep analysis of the contents is the basis for the generation of personalized labels. Huang work [23] explore the issues related to applying extenics methods to build product's resource character, and then the system asks the users to provide the input authority with this system's resource character value for each store. Through the process of assessment, the matching procedure poses

the "buyer's point of view" and then it calculates the matching preference value of each product provided by each store and provides solutions for the selected product, to facilitate a complete deal so both the consumer and producer can get their requirements.

3 The Proposed DPHP Platform

The intuition behind our solution stems from enabling the individuals or groups to control the release of their personal health profiles on a core platform that will store their datasets in a nonproprietary manner to enable the usage of this data in parallel domains, so as to maximize the monetization effort where individual participants benefit from every utilization of their personal health data. However, DPHP is not fully P2P, instead it is a hybrid P2P system like Gnutella [24] there exists a set of nodes connected to each other as seen in Figure (1). A typical application for the DPHP platform involves a genomic research based on bio-banks. Bio-banks are a type of bio-repository that store biological materials like organs, tissue, blood samples, cells, and other body fluids that are containing traces of DNA or RNA. This biological information represents the key resources for a research like genomics and personalized medicine. The research groups and pharmaceutical companies can employ the data stored in the bio-bank for clinical trials, personalization of treatments or research purposes. Bio-banks can employ HSNs to collect genetic or health data from patients and then sharing it with different external parties like – healthcare providers, research and government institutions, and industry. Moreover, DPHP can be utilized as data sharing platform to verify the research output of any health related analytical studies with other dataset representing another random sample of sufferers. Different research groups which carry out the similar research studies can benefit from this feature. However, patients may not be willing to participate in this platform because they are concerned about the privacy of their health profiles, as the data they are going to release can be used against them if it is linked to their real identity. For example, on the basis of their health profiles, health insurance companies can prevent them from participating in specific insurance programs or certain enterprises can refuse to hire them. The emerged privacy considerations have been handled in DPHP by utilizing the collaborative privacy framework which has been proposed in [12-17] to preserve the privacy of the users' health profiles. This approach will give the participants the confidence that the disclosure risk of their health profiles is eliminated.

The basic element in the DPHP is the Expert Agent Execution Server (EAES), which is an execution environment for the expert agents that have been created by the health expert or researcher. An expert-agent is instructed with the required trial along with the query needed to fetch the data to fulfill this trial. Then after, the expert-agent is forwarded to EAES based on the request of the health expert or researcher. The agent can reside in the EAES and acts as a mapper agent which will be responsible for forwarding its worker agents in order to related data aggregators to fetch the data required for the trial. There also exists a set of Aggregator Service Discovery (ASD) which is responsible for maintaining the information regarding different data aggregators.

3.1 System components

As illustrated before, a high level architecture for the DPHP Platform was depicted in figure (1). DPHP consists of different nodes that are connected through internet (it can be a private network as well). DPHP essentially creates a virtual private network even when an underlying network infrastructure is the public internet. Each Aggregator acts as a gateway for gathering anonymized patients' health profiles from different health social networks. As the patient's consent is essential in this process, he/she is notified once the data collection is started. HSN can give certain benefits (like money, prizes, gift brochures... etc.) for the users who have a sustainable rate in participation within each data collection request. A detailed explanation of different nodes is as follows.

ASD (Aggregator Service Discovery).

An ASD is an entity in DPHP that is responsible for maintaining information about the aggregators. The information about the aggregators should include the domain names, IP addresses and data catalogues. The information about related aggregators can be provided when a health expert tells ASD the kind of data required for the trial in-hand. When only a few aggregators are active, one ASD can be utilized for serving such a small group. However, when more aggregators are deployed, a set of ASDs should be distributed in different zones in order to attain a load balancing for the serving of different data collection requests.

EAES (Expert Agent Execution Server).

EAES is a server in DPHP that is provided to the registered health experts in order to host their expert agents that are equipped with the required trials and queries to search for the data needed for each of these trials. Based on the health expert's searching criteria, the expert agent will forward in parallel a pool of worker agents to the relevant aggregators, which in turn will return the required data for the trial. Sandboxing and logging techniques can be utilized to protect both of the execution server and expert-agents from malicious attacks.

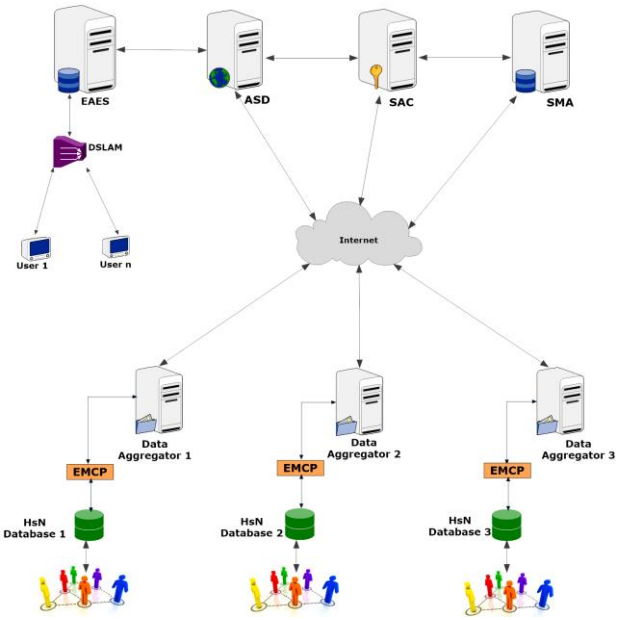


Fig. 1. An overview of *DPHP* Platform

SAC (Security Authority Center).

SAC is a trusted third party in DPHP that is responsible for generating certificates for all aggregators, and managing them. Additionally, SAC is responsible for making security assessment on those authorized aggregators according to the attack and feedback reports which are collected from the participants and the health experts. Then after, SAC submits periodic reports to ASD in order to reflect the updates in the trust ranks of registered aggregators.

SMA (Success Management Authority).

SMA is the authority within DPHP that is responsible for assessing the success criterion for all aggregators. When an aggregator cheating occurs, a health expert can report this to the SMA. After investigation, the success criterion of this aggregator will be downgraded, this in turn diminishes its revenues and the credibility of the data collected from this aggregator. On the other hand, the successful processes will help to amend the success criterion for each aggregator.

Health Expert.

The beneficiary of the DPHP platform who should be a registered expert patient or a researcher running a trial for his/her own. Moreover, the health expert could be a medical research institute or pharmaceutical company enrolled with any EAES before utilizing the facility of submitting task agents and collecting data using the DPHP platform. The health expert can utilize DPHP to search for specific data that is needed for his/her research or trial through an expert agent hosted on EAES. Additionally, the payment for the extracted data is also done through the EAES using a secure e-payment system. Finally, the health expert is also responsible for sending appeals to the SMA for any aggregator cheating that may occur during the trial and/or data collection which is difficult to be detected before the payment. If the cheating is true, the aggregator's success criterion will be degraded, which will result in decreasing the number of worker agents that are being forward there.

3.2 The Search Workflow in DPHP

Based on the proposed framework, the process of enabling the selection and collecting numerous datasets from various aggregators can be described as follows:

1. **Health Expert Requirement Elicitation:** The health expert selects an ASD where he/she has registered as a user in order to create an expert agent. Then after, He/she inputs the query for selecting the dataset that is required for the trial in hand. Moreover, he/she specifies the properties related to the extracted datasets such as price, type, accuracy level, anonymization level, tuples types, no of records, gathering methods and demographics. Finally, he/she also determines the attributes for the potential aggregators, such as the trust rank and success criterion.
2. **Aggregators Selection:** After the health expert dispatches the expert agent to the EAES. The EAES will host this expert agent in order to allow for the completion of its required task. The expert agent divides the required processing along with data query between different primary agents (PA) such that each one of them will be containing one sub-task and one sub-query. These primary agents will be tasked to reside within the qualified aggregators and then forward in parallel a pool of worker agents (WA) to fetch the required data. An aggregator is selected only if its trust rank and success criterion meets the same requirements specified by the health expert. The values for these attributes can be obtained from ASD, SMA, and SAC.
3. **Datasets Assessment:** When the results are returned back by all the worker agents, a second stage of assessment is taken on both properties of datasets and aggregators' trust rank and success criterion. The sorted results are presented back to the health expert by the expert agent.
4. **Negotiation with the Successful Aggregators:** Based on the decision of the health expert, a fewer aggregators will be short listed and selected for negotiation, and then the expert agent will start forwarding negotiation-agents to these selected aggregators. A lot of negotiation models have been proposed and can be utilized for such process [25] . However, in this paper, we will not address this issue.
5. **Payment for Aggregators:** With the successful results of negotiations, one or more aggregators will be favored to collect the dataset from, and then an online secure payment occurs between the expert-agent and

each one of the selected aggregators. Different e-payment models can be utilized for this purpose such as the model proposed in [26].

6. **Feedback from the Health Experts:** After receiving the required dataset from the selected aggregators, the health expert can evaluate the whole process or report the aggregator cheating. The success criterion of such aggregator will be modified based on the feedback from health experts. In addition, during the whole process, in the case of the detection of any attacks from malicious hosts on the primary or worker agents [27], the expert agent at the EAES will report this to SAC, and this will lead to the deterioration of the trust rank for this aggregator. Thus, the number of agents which are being forwarded to such aggregator will be decreased, since the aggregators' selection step takes place before forwarding any of the primary agents there.

4 The Fuzzy Search Model in DPHP Platform

In our framework, we have developed a fuzzy search model that is much more powerful in search than using the conventional matching models when used for research and investigation of unfamiliar, complex, imprecise and ambiguous cases. The proposed model can also be applied to locate multiple datasets and various aggregators based on incomplete or partially inaccurate properties, the returned results by the fuzzy search model are likely based on the subjective relevance. DPHP has easily employed software agents in order to attain parallel and distributed processing. When an expert agent is created and starts running at EAES, it retrieves from ASD a list of aggregators that offer specific datasets needed for the trail that has been specified by its health expert. Then after, the expert agent starts to dispatch a set of primary agents to the selected aggregators. Where, each primary agent forwards multiple of worker agents for querying the metadata of datasets that are offered by the numerous nodes that exist within each registered HSN with a certain aggregator. This metadata involves attributes of each dataset, such as price, type and accuracy level. Each worker agent is responsible for visiting one node within each HSN. Once all the worker agents fulfill their tasks, the primary agents send the results back to the expert agent. Suppose there are hundreds or thousands of nodes which are offering the same kind of datasets. It is unnecessary and even impossible for a health researcher or even a mobile agent to browse all of them. So it is quite necessary and reasonable for the health researcher to find a way to evaluate these nodes and gets the best nodes for further investigation. This assessment process is not only compatible with the human behavior, but also can reduce the network load. Moreover, the number of datasets may be several times more than the number of aggregators, since each aggregator may provide multiple health profiles to the health expert. The health expert should evaluate these datasets and get a short list for the best of them then negotiate with the aggregator for further benefits. The search model in DPHP platform explores the issues of allocating the best and most convenient aggregators to the health expert as well as assessing and refining their datasets, and then returning the best datasets to the health experts. The allocation and assessment are based on a set of predefined selection criteria that are domain specific. Additionally, as most of the real-world situations that can involve constraints that may be imprecisely defined, such as recent datasets, high accuracy and so on, additionally, the common knowledge may be limited to the expert agent. The expert agent should be autonomous enough in order to have the ability to consider these incomplete and imprecisely information. In DPHP platform, we applied the fuzzy rules technologies that have the ability to naturally process incomplete and imprecise information to extract rational results.

Our proposed Fuzzy search model has several features and advantages as it consists of two sequent and correlated stages, the first is the aggregators' selection stage then the datasets assessment stage. The second stage is processed based on the results obtained from the first one. This model can reduce the network load that makes it suitable for an environment where the computing resources are limited. The expert agent can search more nodes and datasets based on the real-time situation and generates more reasonable results.

4.1 Preliminaries: Fuzzy Set and Linguistic Variables

In mathematics, a fuzzy set is different from a crisp set as each element within the fuzzy set has a degree of membership. The membership function is responsible for defining the relationship between a value in the set's domain and its degree of membership [28]. Linguistic variables [29] are variables whose values are not numbers but words or sentences in a natural or artificial language. They are used as a counterpart to the concept of numerical variables. As we mentioned earlier, we have applied fuzzy rules technologies as one of the main building blocks in our fuzzy search model. The fuzzy rule based model [30] consists of a rule base of the following form:

$$if\ V_1\ is\ A_{i1}\ , V_2\ is\ A_{i2}\ , \dots , V_n is\ A_{in}\ , then\ U\ is\ B_i$$

TheV_i's are the antecedent variables and U is the consequent variable. TheA_{ij}'s and B_i's are the fuzzy subsets over the corresponding variable's domain; generally, these subsets represent the linguistic variables. The fuzzy rule based model determines the consequent variable's $U's$ value for a given manifestation of the antecedent variables A_{ij}. This model utilizes principles from utility and fuzzy theories which make such a model straightforward and simple.

Assume a variable x is consisting of a number of attributes:

$$x = \{x_1, x_2, \dots, x_n\} \quad (1)$$

1. For each attribute x_n , calculate its membership level as:

$$A_i = F_i(x_i) \quad (2)$$

where F_i is a semantic function for the attribute x_i.

2. Calculate the units/levels of each attribute as:

$$U_i = V_i(A_i) \quad (3)$$

where V_i is a transfer function that maps the attribute into pre-specified values in a numerical interval i.e. [1,10].

3. Calculate the overall utility of the variable x as:

$$U(x) = \sum w_i U_i \quad (4)$$

where the relative importance assigned for each attribute is represented as a normalized weight w_i such as $\sum w_i = 1$.

4. Calculate the overall membership value of the variable x as:

$$U = F\big(U(x)\big) \quad (5)$$

where F is a transfer function for x. So the overall membership value for a variable $x = \{x_1, x_2, \dots, x_n\}$ in a multi-dimensional space is defined as:

$$U = F\big(\sum w_i V_i\big(F_i(x_i)\big)\big) \quad (6)$$

4.2 Transforming Linguistic Variables using Semantic Function

The Semantic function is responsible for assigning each linguistic attribute into its meaning as a membership value. These values are usually represented as linguistic values, such as very clear, clear, semi-sanitized, sanitized or encrypted. These functions have several features as follows:

- These functions are attribute dependent, i.e. for different linguistic attributes, there may exist different levels for each category. In addition, for the attributes that can be represented as digital values, i.e. price and number of attributes, the semantic functions can use these digital values directly; for the attributes that cannot be represented as digital values directly, i.e. accuracy level and anonymization level, a table should be built that maps these linguistic values into digital values.
- These functions can either classify attribute values into pre-defined number of categories or classify them based on real-time properties of the dataset's metadata. In the first case, the health expert should specify the number of categories that he/she prefers. In the other one, the expert agent summarizes all the information that has been collected from the DPHP platform and then it starts to extract the standard categories based on this information. These standards are dynamic and suitable for this process only.

As the computing resource for the expert agent is limited, we have used a modified version of *LLA* algorithm that was proposed in [31] for the first case described above. Then after, we adopted another algorithm for the latter one.

4.3 Mapping Attributes Using Transfer Function

A Transfer function is responsible for mapping the attribute's membership levels into pre-specified values in a numerical interval i.e. [1, 10]. DPHP makes use of a linear transfer function of the following type:

$$U(A_i^*) = \frac{Max\ A^* - A_i^*}{Max\ A^* - Min\ A^*}\ 0.9 + 1 \quad (7)$$

Or

$$U(A_i^*) = \frac{A_i^* - Min\ A^*}{Max\ A^* - Min\ A^*}\ 0.9 + 1 \quad (8)$$

Where A_i^* represents the average value of current category level. Meanwhile, DPHP uses equation (7) if the function is decreasing with respect to A_i^*, and equation (8) if increasing.

Modified *LLA* Clustering Algorithm

Inputs

Initial values: $X_i(i = 1 \dots n)$

Number of categories: k

Outputs

Clustering Results: $Y_j(j = 1 \dots k)$

1. Select any values $X_{i1}, X_{i2}, X_{i3}, \dots X_{ik}$ from X_i randomly
2. Set an initial starting category $Y_j = X_{ij}(j = 1 \dots k)$
3. Do until the group member is stable

For each $X_i(i = 1 \dots n)$

If $X_i \in [Y_j, Y_{j+1}]$

$D_1 = Distance\ (X_i, Y_j)$

$D_2 = Distance\ (X_i, Y_{j+1})$

If $D_1 < D_2$ then

X_i is in the cluster (category) of Y_j

Else X_i

is in the cluster (category) of Y_{j+1}

End if

End If

End for

Y_j = the average of cluster $Y_j(j = 1 \dots k)$

End Do

Adopted Simple Categorization Algorithm

Inputs

Initial values: $X_i (i = 1 \dots n)$

Fuzzy factor ζ

Outputs

Categories results: $Y_j (j = 1 \dots n)$

1.Sort X_i by ascent or descent to B_i

2.Set the current Categorylevel $= 1$

3.Set item number A in current category level $= 1$

For each $B_x (x = 2 \dots n)$

$$B^* = \frac{1}{A+1} \sum_{m=x-A}^{x} B_m$$

If $\frac{|B^* - B_x|}{B^*} > \zeta$ then

B_x is not in this level

Categorylevel $=$ Categorylevel $+ 1$

A $= 0$

Else B_x is in this level

$Y_x =$ Categorylevel

$A = A + 1$

End if

End for

4.4 Fuzzy Search Model in DPHP Platform

In this paper, the proposed fuzzy search model is executed in three stages: Input, aggregator selection and dataset assessment.

4.4.1 Input.

In this stage, the expert agent collects from the health expert the queries that are needed to retrieve the data which are required for the trial in-hand along with the properties related to the collected datasets and the attributes for the potential aggregators. The health expert's requirements can be further organized into "debatable" requirements and "inalienable" requirements; the "inalienable" requirements are used as the basic conditions in search stage while the "debatable" requirements can be used in the negotiation stage. Moreover, the health expert should select suitable standard categories that will be predefined in the expert agent or learn the health expert's requirements by specifying the relative weights of each attribute and/or property. Finally, the health expert should specify the selection criteria such as the number of aggregators /datasets to be selected or the selection percentage. The expert agent can select the aggregators and evaluate the candidate datasets, and then the negotiation with the appropriate aggregators about their datasets is based on the health expert's requirements.

4.4.2 Aggregators Selection

This stage explores the issues of selecting the appropriate and most potential aggregators to the health expert's requirement in the DPHP platform. Before the start of forwarding any worker agents there, this selection stage is done only over several attributes such as the success criterion, trust rank, and the type of datasets. The success criterion of each aggregator is a value that is determined based on the number of its previous successful processes and the nodes with a low price and accurate health profiles that are affiliated with it. The aggregator which is attracting large number of appropriate nodes from the HSN will get quickly a high success criterion. Those attributes for the aggregator selection stage are stored in the ASD with the domain names, IP addresses and data catalogues for all nodes. After selection, worker agents will be forwarded to those appropriate aggregators for searching in parallel their datasets. In this stage, the processes of selecting aggregators are done in three more steps: aggregator selection, aggregator assessment and aggregator refining.

- Aggregator Selection: in this step, the expert agent queries the ASD' database using the requirements specified by the health expert in order to get the domain names, IP addresses of the correlated aggregators.
- Aggregator Assessment: in this step, the ranking of the aggregators are computed based on our fuzzy rule based model, where the overall membership function is defined as follows :

$$U = F\left(\sum w_i \, V_i\left(F_i(x_i)\right)\right) \text{ where } x = \{S, T, D\}$$

The variable x could be one of the following:

1. S denotes to the success criterion of the aggregator. The aggregator with larger number of previous successful processes and better feedback reports receives a higher value of S. For every successful process, the aggregator will receive a number of success points. Also the health expert can rate the datasets which were gained from the search process. The aggregator can get additional credit points with the positive rating, or miss some credit points if the rating is negative. The information regarding the success criterion of different aggregators is maintained by the SMA.
2. T denotes to the trust rank of the aggregator. In DPHP platform, SAC is the entity which is responsible for making trust assessment on those authorized aggregator according to the attack reports obtained from various parties in DPHP. Then after, SAC periodically reports the updates in the trust ranks of aggregator to ASD. Higher trust rank means higher security level for the aggregator.
3. D denotes to the time required by the aggregator to assemble and deliver the prospective datasets. The type of datasets is quite important to the health expert. It can be long if the health expert is demanding more sophisticated datasets that will require various pre-processing steps in order to be collected and prepared for the delivery. However, the size of the datasets itself is the main impact factor within

the type of datasets variable. Therefore, the aggregators should have a pre-specified datasets types for each of the required processing scopes, and only offer datasets for the health experts in these domains. In DPHP, each aggregator has a table to illustrate the time for delivering the datasets from the nodes in HSN to the health expert, such as Table 1.

Datasets Type (per 400 records)	Numerical Measurements	Pictures	Recorded Signals	Other
Time (hours)	50	100	80	NA

Table 1. : Time Required for Different Datasets Types

Table 1 illustrates the datasets type of aggregator ABC. The required time to collect and prepare 400 records of numerical measurements is 50 hours, for pictures is 100 hours, and for recorded signals is 80 hours. Moreover, this table means that the aggregator ABC only offers datasets for the health experts from these dataset's types. If the health expert demands a dataset that the aggregator doesn't support such as textual data, then the value of D will be set to 0.

- Aggregator Refining: in this step, a list of aggregators addresses are returned to the health expert based on the assessment results and selection criteria that he/she has specified. The selected aggregator in this list must fulfill at least three conditions as follows:

  a) the aggregator that is active
  b) the aggregator in high level
  c) the aggregator that has the $D \neq 0$

  Condition a) ensures that the aggregator is online. Condition b) ensures that the aggregator is "better" than the other aggregators that were not selected. While Condition c) ensures that datasets requirements for the health expert can be met at this aggregator. At the end of the aggregator selection stage, a number of aggregators are returned to the health expert, where he/she can select some/all of these aggregators in the list for a further search process.

4.4.3 Datasets Assessment

In this stage, datasets assessment occurs when all the worker agents send back additional information regarding the datasets, such as the price, accuracy level, anonymization level, tuples types, no of records, gathering method and demographics. Hence another search process will be conducted again over all the gathered properties and the sorted results of appropriate datasets will be presented to the health expert. Upon the health expert decision, the expert agent can now send a new set of worker agents to a selected set of visited aggregators to negotiate for a lower price or more convenient accuracy level. According to the results, the health expert will choose one or more aggregators for data collection and payment.

The datasets assessment stage is similar to the aggregator selection stage but instead of searching the aggregators' attributes, the search process is done over the properties of the various datasets which are offered by the selected aggregators from the previous stage. In this paper, the process of datasets assessment is carried out in two steps: datasets assessment and datasets refining.

- Datasets Assessment: in this step, the ranking of the datasets is computed based on our fuzzy rule based model. In DPHP, the overall membership function is defined as follows :

  $U = F\left(\sum w_i \, V_i\left(F_i(x_i)\right)\right)$ where $x = \{P, D, W, C\}$

  The variable x could be one of the following:

  1. P denotes to the price of the datasets
  2. D denotes to the "no of records" within datasets
  3. W denotes to the anonymization level of the datasets
  4. C denotes to the accuracy level of the datasets

  We have pre-defined several standard categories with different weight for each category. The health expert can either use these pre-defined standard categories or customize the weight of each category based on the real-time properties of the dataset's metadata. The four standard categories that we have defined are as followed:

  - The Category of Price Priority: If the health expert takes the price as the most important factor for search and selection, he/she can select standards in this category. In this category, the price is the main impact factor to be utilized when assessing the datasets, rather than the other properties. The datasets with a lower price can get a higher score. There are three levels in this category: proportional price priority, modest price priority and maximum price priority. Thus, within each level the relative weight of the price variable is increased gradually.
  - The Category of Size Priority: If the health expert wants to get big datasets as much as possible, such that these datasets contain a large number of records, then the "no of records" property is the most important factor for him/her. The datasets with a large number of records can get a higher score. There are three levels in this category: proportional size priority, modest size priority and maximum size priority.
  - The Category of Accuracy Priority: In this category, the health expert prefers more accurate datasets which have been collected by experience patients using modern and well- known medical devices. This category is suitable for healthcare providers and pharmaceutical companies, who want to perform various data analyses on the collected datasets in order to improve the clinical diagnosis and measurements for some medications in treating certain diseases, execute specific clinical trials, and/or

other research purposes. There are also three levels in this category: proportional accuracy priority, modest accuracy priority and maximum accuracy priority.
  - The Category of Balance Priority: In this category, the health expert has no explicit preference. The weights of different properties are similar.
- Datasets Refining: in this step, a sorted list of all datasets is returned to the health expert based on the search process result. The health expert can select some/all of the datasets and negotiate with the aggregators about these datasets in order to attain further benefits.

5 A Case Study on DPHP Platform

In this section, we will present a case study to illustrate the fuzzy search model in DPHP clearly. If we suppose a health expert wants to collect a dataset related to her research. At first, she registers at EAES, and then she creates an expert agent in order to be assigned with the task of collecting the required data for her research. She sets the price and accuracy as "debatable" requirements and other requirements as "inalienable" queries. She prefers a lower price than other properties, so she sets the main factor for the assessment of the datasets to be the category of price priority, where she selects modest price priority as her requirement. The health expert query is shown as follows:

Dataset: Diabetes Measurements
Owner: Older Male Patients
Collection Method: Blood Glucose Meter
Price: <=$1500
Accuracy Level: 7
Dataset Size: 500
Rating Standard: modest price priority
Selection Ratio: Top 25%

Note, the accuracy level is a numerical value within the interval $[1,15]$, and it reflects the degree of the correctness and confidence for each data element within the dataset. Moreover, the selection ratio Top 25% means that she only wants top 25% of the aggregators to be included in the results' list.

5.1 Aggregator Selection

In this stage, the worker agents perform a search process for selecting the appropriate aggregators; the selection is done only over the several attributes that are associated with the aggregators, such as a success criterion, trust rank and type of datasets. The health expert can set the number of aggregators she needs or she can only set the percentage of the aggregators to be selected, i.e. she can select the first 100 or top 25% aggregators and then she starts forwarding the worker agents to these selected aggregators. Assume the expert agent gets a list from ASD with a 100 aggregators that offer the datasets that the health expert requires. After the aggregators selection, the health expert selects top 25% of the aggregators with a better success criterion, higher trust rank and have datasets in the required type of datasets. Then the expert agent sends a set of worker agents to these aggregators in order to get detailed information regarding their offered datasets. The search results are shown in Table 2. These results were extracted based on the selection stage and the requirements that the health expert has specified. All the aggregators that have the same membership value in results were selected (aggregator ID 106 with overall membership value III was also selected). This fuzzy search model is compatible with the human behavior because all these aggregators will look the same for those that will be selected manually by the health expert.

Aggregator ID	Success Criterion	Trust Rank	Dataset Size	Membership Value	Result
22	I	I	532	I	yes
88	II	II	700	III	yes
106	I	II	500	III	yes
135	II	III	720	IV	No
174	V	IV	234	VI	No
201	V	IV	100	VII	No

Table 2. Aggregator Selection Results

5.2 Datasets Assessment

If we assume that half of the selected aggregators offer more than one dataset to the health expert. For example if each aggregator offers five datasets, the health expert will get at least 60 datasets to be manually investigated further. It is impossible for the health expert to investigate 60 datasets in a short-time and consumes unnecessary time in the negotiation process with 12 aggregators. The health expert efforts and time should be consumed efficiently in the clinical trial on her hand. Using DPHP, The health expert should be able to select the best of datasets and then negotiate with the aggregators for further benefits. To illustrate the datasets assessment stage simply, we have used an example of 7 offers. The fuzzy factor ζ in the simple categorization algorithm is set to be $(2\%, 8\%)$ and the number of categories k in the modified *LLA* clustering algorithm is set to be $(5,3)$ for the two properties. In order to compare the results, we have used the transfer function to map the membership levels into pre-specified values in a numerical interval of $[1, 10]$. The results for datasets assessment stage are shown in Table 3.

The results were extracted based on the real-time properties of the datasets' metadata which have been categorized into various levels. From these two tables, we can assure that the results are more appropriate and compatible with the health expert decision making process. The datasets in the same category have no differ-

ence to the health expert. The health expert can freely select the aggregators within any top levels for further negotiation.

Datasets_ID	Price,Accuracy	Simple Categorization		Modified LLA		FinalValue
1	870, 2	I, II	I	I,I	I	9.51
2	970,3	I,III	I	I,I	I	9.43
3	1008,1	I,I	I	I,I	I	9.61
4	1420,1	III,I	I	II,I	I	9.01
5	880,14	I,VI	II	I,III	I	8.86
6	1200,15	II,VI	II	I,III	I	7.53
7	1400,15	IV,VI	II	III,III	II	6.42

Table 3. Datasets Assessment results

6 Conclusions and future work

In this paper we present the proposed core platform which entitled Distributed Platform for Health Profiles (DPHP) that enables individuals or groups to control their personal health profiles and maximize the effort where users benefit from each usage for their personal health profiles. A fuzzy search model based on DPHP was presented and discussed in details. The proposed model is compatible with the health expert decision making process. It aids the health expert in the selection and assessment of the appropriate datasets from a huge pool of distributed datasets that are stored in the personal profiles of health social networks. Multiple attributes and/or properties can be utilized within the proposed fuzzy search model. Clustering algorithms were employed to provide an enhanced feature in the proposed model by extracting the categories of the various properties from the real-time properties of the datasets' metadata, which aids in obtaining dynamic and realistic results for the search process. This model can reduce the network load that makes it suitable for an environment where the computing resources are limited.

Our future research agenda will include extending this model with social recommendation techniques in order to facilitate the preferences' learning for the input stage. Utilizing trust attains the success for selecting the aggregators but a possible new dimension could envision expressing this relation for each user independently without the need for a trusted third party. This would provide a more accurate representation of the trusted aggregator, not influenced as much by the dominant users in the system and business deals. Moreover, in all of the applications, users' trustworthiness is out of interest. Considering malicious user existence would get interesting discussions to grow up.

A more thorough assessment of our model would be useful, such as case studies on a small or large scale. Furthermore, it would be appealing to investigate other innovative applications, which can be used in everyday life, with emphasis on the health profiles.

References

1. Giustini, D.: Web 3.0 and medicine. BMJ **335** (2007) 1273-1274
2. Elkin, N.: How America Searches: Health and Wellness. In: iCrossing (ed.): (2008)
3. M, L.: Online Health: Assessing the risks and opportunity of social and one-to-one media. In: Research, J. (ed.), Vol. 2 (2007)
4. Yanmin, Z., Sye Loong, K., Sloman, M., Lupu, E.C.: A lightweight policy system for body sensor networks. Network and Service Management, IEEE Transactions on **6** (2009) 137-148
5. Lupu, E., Dulay, N., Sloman, M., Sventek, J., Heeps, S., Strowes, S., Twidle, K., Keoh, S.-L., Schaeffer-Filho, A.: AMUSE: autonomic management of ubiquitous e-Health systems. Concurr. Comput. : Pract. Exper. **20** (2008) 277-295
6. Arnst, C.: Health 2.0: Patients as Partners. Business Week (2008)
7. Frost, J., Massagli, M.: PatientsLikeMe the case for a data-centered patient community and how ALS patients use the community to inform treatment decisions and manage pulmonary health. Chronic Respiratory Disease **6** (2009) 225-229
8. Wicks, P.: Parkinson's disease: more non-motor symptoms for younger sufferers. PatientsLikeMe.com (2008)
9. Wicks, P., Massagli, M.P., Wolf, C., Heywood, J.: Measuring function in advanced ALS: validation of ALSFRS-EX extension items. European Journal of Neurology **16** (2009) 353-359
10. Kaye, J., Curren, L., Anderson, N., Edwards, K., Fullerton, S.M., Kanellopoulou, N., Lund, D., MacArthur, D.G., Mascalzoni, D., Shepherd, J.: From patients to partners: participant-centric initiatives in biomedical research. Nature Reviews Genetics **13** (2012) 371-376
11. Fornai, F., Longone, P., Cafaro, L., Kastsiuchenka, O., Ferrucci, M., Manca, M.L., Lazzeri, G., Spalloni, A., Bellio, N., Lenzi, P.: Lithium delays progression of amyotrophic lateral sclerosis. Proceedings of the National Academy of Sciences **105** (2008) 2052-2057
12. Elmisery, A., Botvich, D.: Agent Based Middleware for Private Data Mashup in IPTV Recommender Services. 16th IEEE International Workshop on Computer Aided Modeling, Analysis and Design of Communication Links and Networks. IEEE, Kyoto, Japan (2011)
13. Elmisery, A.M., Botvich, D.: An Agent Based Middleware for Privacy Aware Recommender Systems in IPTV Networks. In: Watada, J., Phillips-Wren, G., Jain, L.C., Howlett, R.J. (eds.): Intelligent Decision Technologies, Vol. 10. Springer Berlin Heidelberg (2011) 821-832
14. Elmisery, A., Botvich, D.: Enhanced Middleware for Collaborative Privacy in IPTV Recommender Services Journal of Convergence **2** (2011) 10
15. Elmisery, A., Botvich, D.: Privacy Aware Recommender Service using Multi-agent Middleware- an IPTV Network Scenario. Informatica **36** (2012)
16. Elmisery, A., Botvich, D.: Multi-agent based middleware for protecting privacy in IPTV content recommender services. Multimed Tools Appl (2012) 1-27
17. Elmisery, A., Botvich, D.: Privacy Aware Obfuscation Middleware for Mobile Jukebox Recommender Services. The

11th IFIP Conference on e-Business, e-Service, e-Society. IFIP, Kaunas, Lithuania (2011)

18. Canny, J.: Collaborative filtering with privacy via factor analysis. Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, Tampere, Finland (2002) 238-245
19. Canny, J.: Collaborative Filtering with Privacy. Proceedings of the 2002 IEEE Symposium on Security and Privacy. IEEE Computer Society (2002) 45
20. Miller, B.N., Konstan, J.A., Riedl, J.: PocketLens: Toward a personal recommender system. ACM Trans. Inf. Syst. **22** (2004) 437-476
21. Raskutti, B., Beitz, A., Ward, B.: A Feature-based Approach to Recommending Selections based on Past Preferences. User Modeling and User-Adapted Interaction **7** (1997) 179-218
22. Robert, D.: Dynamic Document Delivery: Generating Natural Language Texts on Demand. Vol. 0 (1998) 131-131
23. Huang, P.-H.: The Extenics Theory for a Matching Evaluation System. Comput. Math. Appl. **52** (2006) 997-1010
24. Ripeanu, M.: Peer-to-Peer Architecture Case Study: Gnutella Network. (2001)
25. Shen, W., Li, Y., Genniwa, H.H., Wang, C.: Adaptive Negotiation for Agent-Based Grid Computing. In Proceedings of the Agentcities/AAMAS'02 (2002)
26. Kouta, M.M., Rizka, M.M.A., Elmisery, A.M.: Secure e-Payment using Multi-agent Architecture. Computer Software and Applications Conference, 2006. COMPSAC '06. 30th Annual International, Vol. 2 (2006) 315-320
27. Hohl, F.: A protocol to detect malicious hosts attacks by using reference states. Universitätsbibliothek der Universität Stuttgart, Stuttgart (2000)
28. Zadeh, L.A.: Fuzzy sets. Information and Control **8** (1965) 338-353
29. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning—I. Information Sciences **8** (1975) 199-249
30. Mamdani, E.H.: Application of Fuzzy Algorithms for Control of Simple Dynamical Plants. Proc. of IEE **121** (1974) 1585-1588
31. Elmisery, A., Huaiguo, F.: Privacy Preserving Distributed Learning Clustering Of HealthCare Data Using Cryptography Protocols. 34th IEEE Annual International Computer Software and Applications Workshops (COMPSACW), Seoul, South Korea (2010) 140-145