Y. Bhargav *et al*, International Journal of Computer Science and Mobile Applications,
Vol.1 Issue. 5, November- 2013, pg. 18-37 **ISSN: 2321-8363**

Homomorphic Recommendations for Data Packing- A Survey

Y. Bhargav[1], P. Sreenivasa Moorthy[2]

[1]M.Tech. Student, CSE Dept, CMR Institute of Technology, Hyderabad, A.P
Email-id: bhargav.y9959@gmail.com

[2]Associate Professor, CSE Dept., CMR Institute of Technology, Hyderabad, A.P
Email-id: moorthypsm@gmail.com

Abstract

Recommender systems have become an important tool for personalization of online services. Generating recommendations in online services depends on privacy-sensitive data collected from the users. Traditional data protection mechanisms focus on access control and secure transmission, which provide security only against malicious third parties, but not the service provider. This creates a serious privacy risk for the users. In this paper, we aim to protect the private data against the service provider while preserving the functionality of the system. We propose encrypting private data and processing them under encryption to generate recommendations. By introducing a semitrusted third party and using data packing, we construct a highly efficient system that does not require the active participation of the user. We also present a comparison protocol, which is the first one to the best of our knowledge, that compares multiple values that are packed in one encryption. Conducted experiments show that this work opens a door to generate private recommendations in a privacy-preserving manner.

We have developed an approach for privacy-preserving Recommender Systems based on Multi-Agent System technology which enables applications to generate recommendations via various filtering techniques while preserving the privacy of all participants. We describe the main modules of our solution as well as an implemented application based on this approach. This paper also describes various limitations of current recommendation methods and discusses possible extensions that can improve recommendation capabilities and make recommender systems applicable to an even broader range of applications. These extensions include, among others, an improvement of understanding of users and items, incorporation of the contextual information into the recommendation process, support for multcriteria ratings, and a provision of more flexible and less intrusive types of recommendations.

Index Terms- Homomorphic encryption; privacy; recommender systems; secure multiparty computation

1. Introduction

In the last decade, we have experienced phenomenal progress in information and communication technologies. Cheaper, more powerful, less power consuming devices and high bandwidth communication lines enabled us to create a new virtual world in which people mimic activities from their daily lives without the limitations imposed by the physical world. As a result, online applications have become very popular for millions of people. Personalization is a common approach to further improve online services and attract more users. Instead of making general suggestions for the users of the system, the system can suggest personalized services targeting only a particular user based on his preferences. Since the personalization of the services offers high profits to the service providers and poses interesting research challenges, research for generating recommendations, also known as collaborative filtering, attracts attention both from academia and industry.

Social Networks: People use social networks to get in touch with other people, and create and share content that includes personal information, images, and videos. The service providers have access to the content provided by their users and have the right to process collected data and distribute them to third parties. A very common service provided in social networks is to generate recommendations for finding new friends, groups, and events using collaborative filtering techniques [2]. The data required
for the collaborative filtering algorithm is collected from various resources including users' profiles and behaviors.

Online Shopping: Online shopping services increase the likelihood of a purchase by providing personalized suggestions to their customers. To find services and products suitable to a particular customer, the service provider processes collected user data like user preferences and clicklogs.

In all of the above services and in many others, recommender systems based on collaborative filtering techniques that collect and process personal user data constitute an essential part of the service. On one hand, people benefit from online services. On the other hand, direct access to private data by the service provider has potential privacy risks for the users since the data can be processed for other purposes, transferred to third parties without user consent, or even stolen [3].

In this paper, we propose a cryptographic solution for preserving the privacy of users in a recommender system. In particular, the privacy-sensitive data of the users are kept encrypted and the service provider generates recommendations by processing encrypted data. The cryptographic protocol developed for this purpose is based on homomorphic encryption and secure multiparty computation (MPC) techniques. While the homomorphic property is used for realizing linear operations, protocols based on MPC techniques are developed for non-linear operations (e.g. finding the most similar users). The overhead introduced by working in the encrypted domain is reduced considerably by data packing as shown in complexity analysis.

2. Our Contribution

In this work, we consider a scenario where users of an online service receive personalized recommendations, which are generated using collaborative filtering techniques [2]. In this scenario, we aim at protecting the privacy of the users against the service provider by means of encrypting the private data, that is users' ratings, and to generate recommendations in the encrypted domain by running cryptographic protocols, which is an approach similar to [14]–[17]. The output of the cryptographic protocol, as well as the intermediate values in the algorithm, is also private and not accessible to the service provider. It is important to note that while generating recommendations by processing encrypted data is possible, the difficulty lies in realizing efficient privacy-preserving protocols. Our goal is to provide a more efficient privacy-preserving recommender system by improving the state-of-the-art further.

We achieve our goal of having an efficient recommender system as follows.We eliminate the need for active participation of users in the computations by introducing a semitrusted third party, namely the privacy service provider (PSP), who is trusted to perform the assigned tasks correctly, but is not allowed to observe private data. With this construction, the users, who use an applet or a browser plug-in for the service, upload their encrypted data to the service provider and the recommendations are generated by running a cryptographic protocol between the service provider and the PSP, without interacting with the users. As a consequence of this construction, the cryptographic protocol between the service provider and the PSP to generate recommendations has to work on encrypted data only, which makes it impossible to benefit from homomorphic operations as in [16] and [17]. Therefore, we employ alternative ways of processing encrypted data like secure multiplication and decryption protocols, which introduce a significant amount of additional computational overhead to the system. We reduce the computational and communication cost significantly by data packing, a construction similar to [18] and [19], in which several numerical values are packed in a compact way prior to encryption. We also present a cryptographic protocol that compares encrypted and packed data, which is, to the best of our knowledge, the only algorithm existing in the literature. We analyze the performance of our proposal by conducting experiments on a dataset with 10 000 people and their ratings for 1000 items. We compare two versions of our proposal, with and without data packing, in terms of bandwidth and runtime.

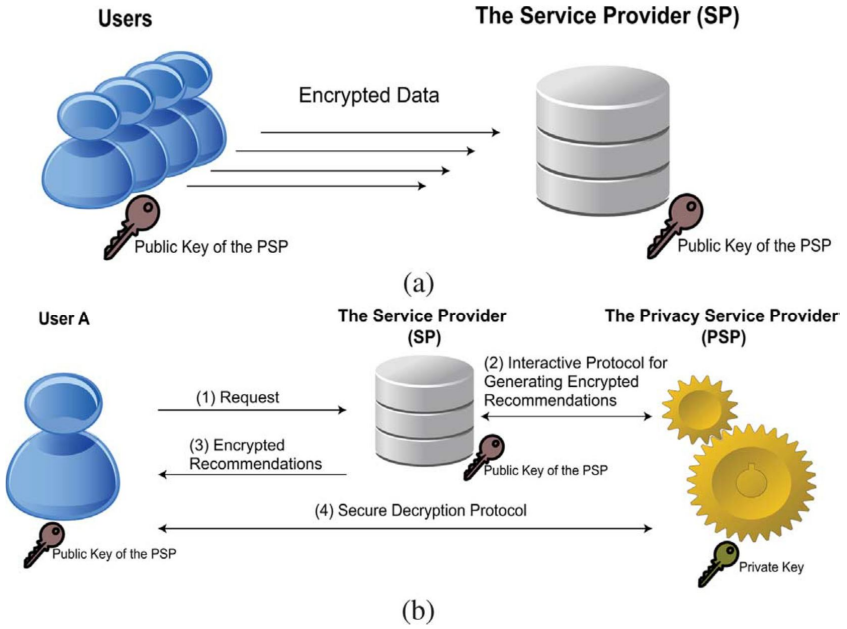

Fig. 1. System model of generating private recommendations. (a) Encrypted database construction; (b) generating private recommendations.

The Privacy Service Provider (PSP) is a semitrusted third party who has a business interest in providing processing power and privacy functionality. The PSP has private keys for the Paillier and the DGK cryptosystems.

Users are the customers of the service provider. Based on their preferences, in the form of ratings, the service provider generates recommendations for them.

The goal of our protocol is to hide any piece of information that may harm the privacy of users. That is, for a particular user, rating vectors, computed similarity values, results of comparing similarity values to a public threshold, and generated recommendations are all kept secret from the SP, the PSP, and all other users. Among these, only the generated recommendations and the number of users , whose ratings are considered for generating recommendations, will be revealed to the user who asks for recommendations. Our protocol consists of two phases as illustrated in Fig. 1: (a) construction of the encrypted database and (b) generating recommendations.

3. Literature survey

In the last decade, we have experienced phenomenal progress in information and communication technologies. Cheaper, more powerful, less power consuming devices and high bandwidth communication lines enabled us to create a new virtual world in which people mimic activities from their daily lives without the limitations imposed by the physical world. As a result, online applications have become very popular for millions of people.

Personalization is a common approach to further improve online services and attract more users. Instead of making general suggestions for the users of the system, the system can suggest personalized services targeting only a particular user based on his preferences. Since the personalization of the services offers high profits to the service providers and poses interesting research challenges, research for generating recommendations, also known as collaborative filtering, attracts attention both from academia and industry.

The techniques to generate recommendations for users strongly rely on information gathered from the user. This information can be provided by the user he as in profiles or the service provider can observe users' actions, such as click logs. On one hand, more user information helps the system to improve the accuracy of the recommendations. On the other hand, the information on the users creates a severe privacy risk since there is no solid guarantee for the service provider not to misuse the users' data. It is often seen that whenever a user enters the system, the service provider claims the ownership of the information provided by the user and authorizes itself to distribute the data to third parties for its own benefits.

As an example, consider pay-TV boxes. A small box purchased by the user provides high quality broadcasting with several interesting features like recording programs. Companies in this field also suggest programs and movies that they think their customers may like. In order to make useful recommendations to their customers, the small box observes the user behavior: it records the programs watched, the duration spent in front of the TV and so on. The information gathered by the box is then sent to a server and processed to deduce meaningful information about users. It is obvious that this system can be used for harming the user's privacy.

In Canny proposes a system where the private user data is encrypted and recommendations are generated by applying an iterative procedure based on the conjugate gradient algorithm. The algorithm computes a characterization matrix of the users in a subspace and generates recommendations by calculating re-projections in the encrypted domain. Since the algorithm is iterative, it takes many rounds for convergence and in each round users need to participate in an expensive decryption procedure which is based on a threshold scheme where a significant portion of the users are assumed to be online and honest. The output of each iteration, which is the characterization matrix, is available in clear. In Canny proposes a method to protect the privacy of users based on a probabilistic factor analysis model by using a similar approach. While Canny works with encrypted user data, Polat and Du suggest protecting the privacy of users by using randomization techniques. In their paper, they blind the user data with a known random distribution assuming that in aggregated data this randomization cancels out and the result is a good estimation of the intended outcome. The success of this method highly depends on the number of users participating in the computation since for the system to work, the number of users need to be vast. This creates a trade-off between accuracy/correctness of the recommendations and the number of users in the system. Moreover, the outcome of the algorithm is also available to the server who may constitute a privacy threat to the users. Finally, the randomization techniques are believed to be highly insecure.

Generating Recommendations

A centralized system for generating recommendations is a common approach in e-commerce applications. To generate recommendations for user A, the server follows a two-step procedure. In the first step, the server searches for users similar to user A. Each user in the system is represented by a preference vector which is usually composed of ratings for each item within a certain range. Finding similar users is based on computing similarity measures between users' preference vectors. Pearson correlation is a common similarity measure for two users with preference vectors VA = (v(A,0), . . . , v(A,M−1)) and VB = (v(B,0), . . . , v(B,M−1)), respectively, where M is the number of items and $\bar{v}$ represents the average value of the vector v. Once the similarity measure for each user is computed, the server proceeds with the second step. In this step, the server chooses those L users with similarity values above a threshold δ and averages their ratings. These average ratings are then presented as *recommendations* to user A.

In e-commerce applications the number of items offered to users is usually in the order of hundreds or thousands. Apart from many smart ways of determining the likes and dislikes of users for the items, we assume the users are asked to rate the items explicitly with integer values in the range of [0, K]. This rating matrix is usually highly sparse, meaning that most of the items are not rated. Finding similar users in a sparse dataset can easily lead the server to generate inaccurate recommendations. To cope with this problem, one approach is to introduce a small set of items that is rated by most users. Such a base set can be explicitly given to the users or implicitly chosen by the server from the most commonly rated items. Given such a small set of items that is rated by most users, the server can compute similarities between users more confidently, resulting in more accurate recommendations. Therefore, we assume that the user preference vector V is split into two parts: the first part consists of R elements that are rated by most of the users and the second part contains M − R sparsely rated items that the user would like to get recommendations.

Cryptographic Primitives and Security Model

We use encryption to protect user data against the service provider and other users. A special class of cryptosystems, namely homomorphic cryptosystems, allows us to process data in the encrypted form. We choose the Paillier cryptosystem as it is *additively homomorphic* meaning that the product of two encrypted values [a] and [b], (where [·] denotes the encryption function), corresponds to a new encrypted message whose decryption yields the sum of a and b as [a] · [b] = [a + b]. As a consequence of the additive homomorphism, any cipher text [m] raised to the power of a public value c corresponds to the multiplication of m and c in the encrypted domain: [m]c = [m · c]. In addition to the homomorphism property, the Paillier cryptosystem is semantically secure implying that each encryption has a random element that results in different cipher texts for the same plain text.

As a part of a cryptographic protocol introduced we use another additively homomorphic and semantically secure encryption scheme, DGK. The DGK cryptosystem is used to replace the Paillier cryptosystem in a subprotocol, for reasons of efficiency. For the same level of security, DGK has a much smaller message space compared to the Paillier cryptosystem and thus, encryption and decryption operations are more efficient than under Paillier. We use the semi-honest security model, which assumes that all players follow the protocol steps but are curious and thus keep all messages from previous and current steps to extract more information than they are allowed to have. Our protocol can be adapted to the active attacker model by using the ideas with additional overhead. [1]

Conventional cryptographic protocols deal with the problem of protecting some private information from an unauthorized third party that otherwise could modify or have access to the information. When the privacy must be preserved not only against a third party, but also against the parties that participate in the protocol where the inputs are shared, secure multiparty computation constructions can be used. On the other hand, when the problem requires processing noisy signals consisting of a large number of samples, typical multiparty computation protocols become too costly in terms of computation and communication complexity.

This is the context in which the emerging field of signal processing in the encrypted domain arose. This discipline tries to address the problem of processing signals in untrusted environments, where not only the communication channel between parties is insecure, but also the parties that perform the computation are not trusted.

In this privacy preserving computation framework, several proposals have been recently issued to implement primitives like secure access to encrypted databases, transcoding of an encrypted signal without prior decryption, or basic problems in computational geometry, such as computing scalar products or solving the point inclusion problem for the 2-dimensional case.

The point inclusion problem refers to deciding whether a point lies in a certain spatial region. It is related to point location in computational geometry, which has been investigated for two-dimensional spaces for more than twenty years optimizing the algorithms for achieving subpolynomial search time and storage. For more than two dimensions, the point location problem is still open, except for the case of arrangements of hyperplanes, convex subdivisions, especially convex polytopes, and other subdivisions that allow efficient point location. Point inclusion is an underlying problem in many common signal processing applications that must be run in untrusted environments; however, it rarely deals with 2-dimensional signals, but with multidimensional ones. For instance, in the case of biometric authentication, the biometric data a feature vector embodying a point in a multidimensional space that is presented by an individual must be matched with some template represented by a region of acceptance in the space that is held by a server, but both parties do not want to disclose their respective inputs to the other party.

Applications

There are various applications where a practical and secure point inclusion protocol is required. In the following, we list some application fields.

Biometrics: The most evident application of biometrics is authentication. Here, the server has information regarding the biometrics of a person. Due to its fuzziness, the region of acceptance is modeled as a convex polytope in the features space. The user presents her features as a feature vector, and both parties run a secure point inclusion protocol for determining the correctness of the user's claimed identity. In this process, the biometric features of the client are protected from the server, and the region of acceptance is not disclosed to the user. Furthermore, the whole interaction consists of encrypted values, thereby protecting the information against an eavesdropper. Comparing this method with the typical Helper Data Systems employed in biometric authentication, the complexity of our protocol is higher, but its main advantage is its flexibility, as it allows performing fine grained adjustments of the detection boundary.

Classification: The point inclusion problem with a convex polytope can be regarded as a classification problem. Here, the spatial region is interpreted as a fusion of linear classifiers, each one represented by one of the hyper planes that form the polytope boundary. Thus, the developed protocol implements a secure classifier. The case of hyperellipsoids corresponds to a one-layer RBF (Radial Basis Function) network with threshold activation function.

Database queries: The developed point inclusion protocol can also find an application in non-orthogonal database queries, where a query, represented as a convex region in the measurable terms space, is matched with an entry, represented by a vector of terms. In this case, the query is not revealed to the database server, and the server can keep the entries secret until they match a query.

Positioning: If the protocol is restricted to two or three dimensions, it can be applied to the problem of secure positioning. Here, a party wants to check whether one particular location is inside a region whose definition is owned by another party, but neither of them wants to disclose their own data to the other party. The work is a typical example of a secure positioning application in a pervasive sensor network, where a user wants to know if his current position is being sensed, but the monitoring party does not want to disclose the sensing area.

Watermarking/Fingerprinting: Classic symmetric watermarking and fingerprinting schemes require disclosure of the embedding key during detection. In case the party performing watermark detection is malicious, it can use the key to remove a watermark. Thus, traditional symmetric watermark detectors are not applicable in this case. The secure point inclusion protocol can be applied in a secure watermark detector, where the detection region is a convex polytope in a multidimensional space. This makes it possible to run the detection protocol without disclosing either the detection region to the party that presents

neither the possibly watermarked work, nor this work to the party that owns the description of the detection region.

To the best of our knowledge, the only proposal for solving the problem of point inclusion through secure two-party computation was presented by Atallah and Du for a two dimensional problem where the region is a convex polygon. The authors develop two primitives, namely a protocol for privately computing the scalar product of two values, and a vector dominance protocol that privately tests whether all components of one vector are greater than the components of another vector. The latter protocol is based on several parallel executions of Yao's millionaires protocol. Finally, they require a method of equality testing. The protocol by Atallah and Du has been recently used for privately determining the positioning on the sensing area of a pervasive sensor network.

Atallah and Du's solution has several drawbacks. The first problem is related to their protocol for privately computing the scalar product. As pointed out, it does not preserve privacy. With a simple attack one of the parties can, with a probability close to 1, retrieve the private input of the other party after a single execution of the protocol. The second drawback is the inefficiency of their vector dominance protocol, as it involves several executions of Yao's millionaires protocol. Finally, the protocol they propose for equality testing only works when using a commutative deterministic encryption, which cannot achieve semantic security.

In multiparty computation (MPC), we consider a number of players P1…. Pn, who initially each hold inputs x1……xn, and we then want to securely compute some function f on these inputs, where f(x1……. xn) = (y1……. yn), such that Pi learns yi but no other information. This should hold, even if players exhibit some amount of adversarial behavior. The goal can be accomplished by an interactive protocol that the players execute. Intuitively, we want that executing is equivalent to having a trusted party T that receives privately xi from Pi, computes the function, and returns yi to each Pi 1. With such a protocol we can in principle solve virtually any cryptographic protocol problem. The general theory of MPC was founded in the late 80-ties. The theory was later developed in several ways.

Despite the obvious potential that MPC has in solving a wide range of problems, we have seen virtually no practical applications of MPC in the past. This is probably in part due to the fact that direct implementation of the first general protocols would lead to very inefficient solutions.

Another factor has been a general lack of understanding in the general public of the potential of the technology. A lot of research has gone into solving the efficiency problems, both for general protocols and for special types of computations such as voting.

A different line of research has had explicit focus on a range of economic applications, which are particularly interesting for practical use. This approach was taken, for instance, by two research projects that the authors of this paper have been involved in: SCET (Secure Computing, Economy and Trust) and SIMAP (Secure Information Management and

Processing), which has been responsible for the practical application of MPC described in this paper. In the economic field of mechanism design the concept of a trusted third party has been a central assumption since the 70's.

Ever since the field was initiated it has grown in momentum and turned into a truly cross disciplinary field. Today, many practical mechanisms require a trusted third party and it is natural to consider the possibility of implementing such a party using MPC. In particular, we have considered: Various types of auctions that involves sealed bids for different reasons. The most well-known is probably the standard highest bid auction with sealed bids, however, in terms of turnover another common variant is the so called double auction with many sellers and buyers.

This auction handles scenarios where one wants to find a fair market price for a commodity given the existing supply and demand in the market. Benchmarking, where several companies want to combine information on how their businesses are running, in order to compare themselves to best practice in the area. The benchmarking process is either used for learning, planning or motivation purposes. This of course has to be done while preserving confidentiality of companies' private data.

When looking at such applications, one finds that the computation needed is basically elementary arithmetic on integers of moderate size, say around 32 bits. More concretely, quite a wide range of the cases require only addition, multiplication and comparison of integers. As far as addition and multiplication is concerned, this can be handled quite efficiently by well-known generic MPC protocols. What they really do is actually operations modulo some prime p, because the protocols are based on secret sharing over Zp. But by choosing p large enough compared to the input numbers, we can avoid modular reductions and get integer addition and multiplication.

Application Scenario

In this section we describe the practical case in which our system has been deployed. In preliminary plans for this scenario and results from a small-scale demo were described. In Denmark, several thousand farmers produce sugar beets, which are sold to the company Danisco, the only sugar beets processor on the Danish market. Farmers have contracts that give those rights and obligation to deliver a certain amount of beets to Danisco, who pay them according to a pricing scheme that is an integrated part of the contracts. These contracts can be traded between farmers, but trading has historically been very limited and has primarily been done via bilateral negotiations.

In recent years, however, the EU drastically reduced the support for sugar beet production. This and other factors meant that there was now an urgent need to reallocate contracts to farmers where productions pay best. It was realized that this was best done via a nation-wide exchange, a double auction.

Market Clearing Price Details of the particular business case can be found. Here, we briefly summarize the main points while more details on the actual computation to be done are given later. The goal is to find the so called market clearing price (MCP), which is a price per unit of the commodity that is traded. What happens is that each buyer specifies, for each potential price, how much he is willing to buy at that price, similarly sellers say how much they are willing to sell at each price.4 All bids go to an auctioneer, who computes, for each price, the total supply and demand in the market. Since we can assume that supply grows and demand decreases with increasing price, there is a price where total supply equals total demand, and this is the price we are looking for. Finally, all bidders who specified a non-zero amount to trade at the market clearing price get to sell/buy the amount at this price.

Privacy of Bids using Secure MPC A satisfactory implementation of such an auction has to take some security concerns into account: Bids clearly reveal information, e.g., on a farmer's economic position and his productivity, and therefore farmers would be reluctant to accept Danisco acting as auctioneer, given its position in the market. Even if Danisco would never misuse its knowledge of the bids in the ongoing renegotiations of the contracts (including the pricing scheme), the mere fear of this happening could affect the way farmers bid and lead to a suboptimal result of the auction. On the other hand, the entitled quantities in a given contract are administrated by Danisco (and adjusted frequently according to the EU administration) and in some cases the contracts act as security for debt that farmers have to Danisco. Hence running the auction independently of Danisco is not acceptable either. Finally, the solution of delegating the legal and practical responsibility by paying e.g. a consultancy house to be the trusted auctioneer would have been a very expensive solution. [3]

In the last decade biometric identification and authentication have increasingly gained importance for a variety of enterprise, civilian and law enforcement applications. Examples vary from fingerprinting and iris scanning systems, to voice and face recognition systems, etc. Many governments have already rolled out electronic passports and IDs that contain biometric information (e.g., image, fingerprints, and iris scan) of their legitimate holders. In particular it seems that facial recognition systems have become popular aimed to be installed in surveillance of public places, and access and border control at airports to name some. For some of these use cases one requires online search with short response times and low amount of online communication.

Moreover, face recognition is ubiquitously used also in online photo albums such as Google Picasa and social networking platforms such as Facebook which have become popular to share photos with family and friends. These platforms support automatic detection and tagging of faces in uploaded images. Additionally, images can be tagged with the place they were taken. The widespread use of such face recognition systems, however, raises also privacy risks since biometric information can be collected and misused to profile and track individuals against their will. These issues raise the desire to construct privacy-preserving face recognition systems.

In the most recent proposal for privacy-preserving face recognition the authors use the standard and popular Eigenface recognition algorithm and design a protocol that performs operations on encrypted images by means of homomorphic encryption schemes, more concretely, Pailler as well as a cryptographic protocol for comparing two Pailler-encrypted values based on the Damg°ard, Geisler cryptosystem). They demonstrate that privacy-preserving face recognition is possible in principle and give required choices of parameter sizes to achieve a good classification rate. However, the proposed protocol requires O(logN) rounds of online communication as well as computationally expensive operations on homomorphically encrypted data to recognize a face in the database of N faces. Due to these restrictions, the proposed protocol cannot be deployed in practical large-scale applications. In this paper we address this aspect and show that one can do better w.r.t. efficiency. Basically one can identify two approaches for secure computation: the first approach is to perform the required operations on encrypted data by means of homomorphic encryption. The other approach is based on Garbled Circuit (GC) `a la Yao the function to be computed is represented by a garbled circuit i.e., the inputs and the function are encrypted ("garbled"). Then the client obliviously obtains the keys corresponding to his inputs and decrypts the garbled function. Homomorphic Encryption requires low communication complexity but huge round and computation complexity whereas GC has low online complexity (rounds, communication and computation) but large offline communication complexity. We present a protocol for privacy-preserving face recognition based on a hybrid protocol which combines the advantages of both approaches. Additionally, we give a protocol which is based on GC only. [5]

In recent years privacy preserving processing of private signals, usually referred to as signal processing in the encrypted domain (s.p.e.d.), has received an increasing interest due to its possible application to the processing of sensitive data, such as biometric data, biomedical signals, user preferences, etc. where two or more parties are interested to cooperate for obtaining a common result based on their private inputs without revealing them to each other. The main techniques s.p.e.d. protocols are built upon are Homomorphic Encryption (HE) and Garbled Circuits (GC), both belonging to the wider field of multiparty computation. In signal processing applications, we are often interested in a particular case of the above scenario wherein a party C seeks the cooperation of another party S to perform a computational task (Secure Two-Party Computation - STPC).

An important operation that cannot be performed easily in a privacy preserving setting is the integer division between secrets, a commonly and daily used mathematical operation; think for instance to the computation of normalized moments, signal-to-noise ratios, likelihood ratios, etc. According to the secrecy requirements applying to the numerator and denominator of the division, several cases can be distinguished. In the following we will refer to them by differentiating between public, private and secret values. We say that a value is public if it is known to both C and S, while private values are those that are known to one of the parties and must be kept secret to the other. Finally, we say that a value is secret if none of the parties is allowed to know it. This is usually the case for intermediate values of larger computational tasks that cannot be revealed to any of the parties to avoid leakage of information. Particular attention has to be paid to the privacy requirements regarding the denominator. It is quite common, in fact, to assume that the value of the denominator has to

be kept secret but the minimum number of bits necessary to represent it (let us call it `) is known, since this assumption allows for more efficient solutions1. In the following, we will refer to this situation as a division by a constrained denominator.

The problem of computing the division in a privacy preserving scenario has been addressed in the past years in several settings, mainly focusing on homomorphic encryption. In his PhD dissertation, Toft proposes a method for private modulo reduction (and hence division) with public modulo in a constant number of rounds. The protocol is based on binary reduction and comparisons (where in this case ` is the numerator bitlength). The protocol is also extended to the case of secret modulo by using a secure comparison protocol. The authors propose a statistically secure division protocol with public modulo based on blinding and a new obfuscation protocol that is more efficient than the one presented. Guajardo et al. present a modulo reduction protocol that is secure against malicious adversaries. The protocol computes the modulo between a secret value and public modulo with a communication complexity of O(T) bits where T is the bitsize of the RSA-modulus and ` is the bit-length of the modulo.

Veugen presents several protocols for exact and approximate division addressing both the cases of public and private constrained denominator, including an improvement. With regard to exact division, two protocols are presented: the first one computes the division with a public denominator den and is based on binary search (it requires log2 den secure comparisons), but can be extended to the case of a private constrained denominator (log2 den comparisons and O(d) secure multiplications required); the second solution is based on blinding but can be used only with public denominator. Regarding the solutions for approximate division they are provided for the case of public denominator and private constrained denominator. [6]

Information Filtering (IF) systems aim at countering information overload by extracting information that is relevant for a given user out of a large body of information available via an information provider. In contrast to Information Retrieval (IR) systems, where relevant information is extracted based on search queries, IF architectures generate personalized information based on user profiles containing, for each given user, personal data, preferences, and rated items. The provided body of information is usually structured and collected in provider profiles. Filtering techniques operate on these profiles in order to generate recommendations of items that are probably relevant for a given user, or in order to determine users with similar interests, or both. Depending on the respective goal, the resulting systems constitute Recommender Systems, Matchmaker Systems, or a combination thereof. The aspect of privacy is an essential issue in all IF systems: Generating personalized information obviously requires the use of personal data. Users can be expected to be less reluctant to provide personal information if they trust the system to be privacy-preserving with regard to personal data, according to surveys indicating major privacy concerns of users in the context of Recommender Systems and e-commerce in general. Similar considerations also apply to the information provider, who may want to control the dissemination of the provided information, and to the provider of the filtering techniques, who may not want the

details of the utilized filtering algorithms to become common knowledge. A privacy-preserving IF system should therefore balance these requirements and protect the privacy of all parties involved in a multilateral way, while addressing general requirements regarding performance, security and quality of the recommendations as well. The following section lists some approaches with similar goals, but none of these provide a generic approach in which the privacy of all parties is preserved.

There is a large amount of work in related areas, such as Private Information Retrieval, Privacy-Preserving Data Mining, and other privacy-preserving protocols, most of which is based on Secure Multi-Party Computation. We have ruled out Secure Multi-Party Computation approaches mainly because of their complexity, and because the algorithm that is computed securely is not considered to be private in these approaches.

Various enforcement mechanisms have been suggested that are applicable in the context of privacy-preserving Information Filtering, such as enterprise privacy policies or hippocratic databases, both of which annotate user data with additional meta-information specifying how the data is to be handled on the server side. These approaches ultimately assume that the provider actually intends to protect the privacy of the user data, and support him in this regard, but they are not intended to prevent the provider from acting in a malicious manner. Trusted computing aims at realizing a trusted systems by increasing the security of open systems to a level comparable with the level of security that is possible in closed systems. It is based on a combination of tamper-proof hardware and various software components. Some example applications, including peer-to-peer networks, distributed firewalls, and distributed computing in general, are listed.

There are some approaches for privacy-preserving Recommender Systems based on distributed collaborative filtering, in which recommendations are generated via a public model aggregating the distributed user profiles without containing explicit information about user profiles themselves. This is achieved via Secure Multi-Party Computation, or via random perturbation of the user data. In various approaches are integrated within a single architecture. In an agent-based approach is described in which user agents representing similar users are discovered via a transitive traversal of user agents. Privacy is preserved through pseudonymous interaction between the agents and through adding obfuscating data to personal information. More recent related approaches are described.

In an agent-based architecture for privacy-preserving demographic filtering is described which may be generalized in order to support other kinds of filtering techniques. While in some aspects similar to our approach, this architecture addresses at least two aspects inadequately, namely the protection of the filter against manipulation attempts, and the prevention of collusions between the filter and the provider.

Privacy Preserving Information Filtering

We identify three main abstract entities participating in an information filtering process within a distributed system: A user entity, a provider entity, and a filter entity. Whereas in

some applications the provider and filter entities explicitly trust each other, because they are deployed by a single party, our solution is applicable more generically because it does not require any trust between the main abstract entities. In this paper, we focus on aspects related to the information filtering process itself, and omit all aspects related to information collection and processing, i.e. the stages in which profiles are generated and maintained, mainly because these stages are less critical with regard to privacy, as they involve fewer different entities.

Requirements

Our solution aims at meeting the following requirements with regard to privacy:

User Privacy: No linkable information about user profiles should be acquired permanently by any other entity or external party, including other user entities. Single user profile items, however, may be acquired permanently if they are unlinkable, i.e. if they cannot be attributed to a specific user or linked to other user profile items. Temporary acquisition of private information is permitted as well. Sets of recommendations may be acquired permanently by the provider, but they should not be linkable to a specific user. These concessions simplify the resulting protocol and allow the provider to obtain recommendations and single unlinkable user profile items, and thus to determine highly requested information and optimize the offered information accordingly.

Provider Privacy: No information about provider profiles, with the exception of the recommendations, should be acquired permanently by other entities or external parties. Again, temporary acquisition of private information is permitted. Additionally, the propagation of provider information is entirely under the control of the provider. Thus, the provider is enabled to prevent e.g. the automatic large-scale extraction of information.

Filter Privacy: Details of the algorithms applied by the filtering techniques should not be acquired permanently by any other entity or external party. General information about the algorithm may be provided by the filter entity in order to help other entities to reach a decision on whether to apply the respective filtering technique.

In addition, general requirements regarding the quality of the recommendations as well as security aspects, performance and broadness of the resulting system have to be addressed as well. While small trade-offs may be acceptable, the resulting system should reach a level similar to regular Recommender Systems with regard to these requirements.

The basic idea for realizing a protocol fulfilling the privacy-related requirements in Recommender Systems is suggested by allowing the temporary acquisition of private information: User and provider entity both propagate the respective profile data to the filter entity. The filter entity provides the result information, and subsequently deletes all private information, thus fulfilling the requirement regarding permanent acquisition of private information. The entities whose private information is propagated have to be certain that the

respective information is actually acquired temporarily only. Trust in this regard may be established in two main ways:

Trusted Software: The respective entity itself is trusted to remove the respective information as specified.

Trusted Environment: The respective entity operates in an environment that is trusted to control the communication and life cycle of the entity to an extent that the removal of the respective information may be achieved regardless of the attempted actions of the entity itself. Additionally, the environment itself is trusted not to act in a malicious manner (e.g. it is trusted not to acquire and propagate the respective information itself).

In both cases, trust may be established in various ways. Reputation-based mechanisms, additional trusted third parties certifying entities or environments or trusted computing mechanisms may be used. Our approach is based on a trusted environment realized via trusted computing mechanisms, because we see this solution as the most generic and realistic approach.

We are now able to specify the abstract information filtering protocol as shown: The filter entity deploys a Temporary Filter Entity (TFE) operating in a trusted environment. The user entity deploys an additional relay entity operating in the same environment.

Through mechanisms provided by this environment, the relay entity is able to control the communication of the TFE, and the provider entity is able to control the communication of both relay entity and the TFE. Thus, it is possible to ensure that the controlled entities are only able to propagate recommendations, but no other private information. In the first stage, the relay entity establishes control of the TFE, and thus prevents it from propagating user profile information. User profile data is propagated without participation of the provider entity from the user entity to the TFE via the relay entity. In the second stage, the provider entity establishes control of both relay and TFE, and thus prevents them from propagating provider profile information. Provider profile data is propagated from the provider entity to the TFE via the relay entity. In the third stage, the TFE returns the recommendations via the relay entity, and the controlled entities are terminated. Taken together, these steps ensure that all private information is acquired temporarily only by the other main entities. The problems of determining acceptable queries on the provider profile and ensuring unlinkability of the result information are discussed in the following section.

Our approach requires each entity in the distributed architecture to have the following five main abilities: The ability to perform certain well-defined tasks such as carrying out a filtering process with a high degree of autonomy, i.e. largely independent of other entities e.g. because the respective entity is not able to communicate in an unrestricted manner, the ability to be deployable dynamically in a well-defined environment, the ability to communicate with other entities, the ability to achieve protection against external manipulation attempts, and the ability to control and restrict the communication of other entities.

MAS architectures are an ideal solution for realizing a distributed system containing all features outlined above, because they provide agents as entities that are actually characterized by autonomy, mobility and the ability to communicate, as well as agent platforms as environments providing means to realize the security of agents.

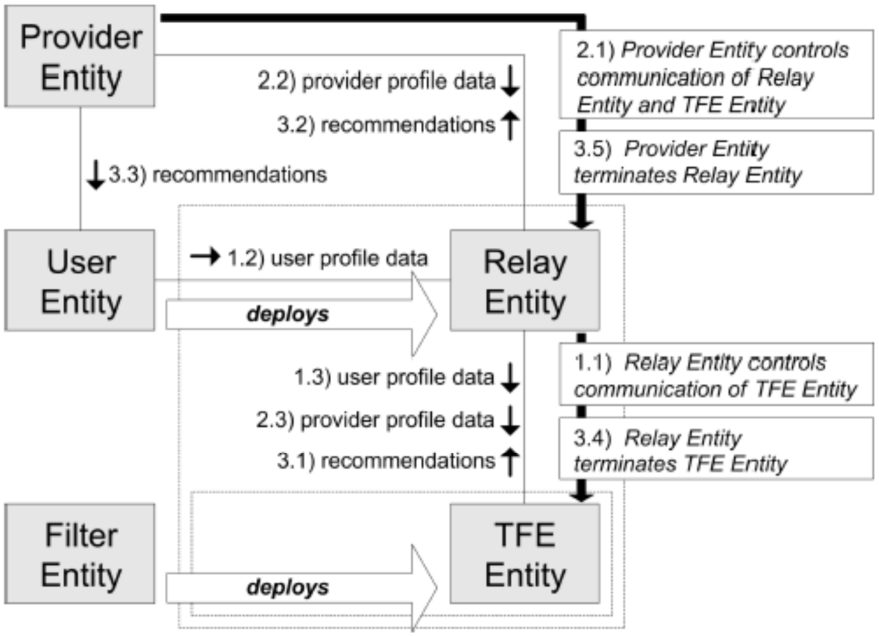

Figure 1: The abstract privacy-preserving information filtering protocol. All communication across the environments indicated by dashed lines is prevented with the exception of communication with the controlling entity.

In this context, the issue of malicious hosts, i.e. hosts attacking agents, has to be addressed explicitly. Additionally, existing MAS architectures generally do not allow agents to control the communication of other agents. It is possible, however, to expand MAS architecture and to provide designated agents with this ability. For these reasons, our solution is based on a FIPA-compliant MAS architecture. The entities introduced above are mapped directly to agents, and the trusted environment in which they exist is realized as agent platforms.[7]

In the last decade, we have experienced phenomenal progress in information and communication technologies. Cheaper, more powerful, less power consuming devices and high bandwidth communication lines enabled us to create a new virtual world in which people mimic activities from their daily lives without the limitations imposed by the physical world. Online shopping, banking, communicating and much more have be- come common for millions of people.

Personalization is a common approach to attract even more people to online services. Instead of making general suggestions for the users of the system, the system can suggest personalized services targeting only a particular user based on his preferences. Since the personalization of the services offers high profits to the service providers and poses interesting research challenges, research for generating recommendations, also known as collaborative filtering, attracts attention both from academia and industry.

The techniques for generating recommendations for users strongly rely on the information gathered from the user. This information can be provided by the user he as in profiles or the service provider can observe user's actions like click logs. On one hand, more information on the user helps the system to improve the accuracy of the recommendations. On the other hand, the information on the users creates a severe privacy risk since there is no solid guarantee for the service provider not to misuse the user's data. It is often seen that whenever a user enters the system, the service provider claims the ownership of the information provided by the user and authorizes itself to distribute the data to third parties for its own benefits.

Privacy-Preserving Recommender System

In this section we propose a protocol based on additively homomorphic encryption schemes and MPC techniques. In particular the service provider, i.e. the server, receives the encrypted rating vector of user A and sends it to the other users in the system who can then compute the similarity value on their own by using the homomorphism property of the encryption scheme. Once the users compute the similarity values, they are sent to the server. After that, the server and user A runs a protocol to determine the similarity values that are above a threshold δ. The server, being unaware of the number users with a similarity value above a threshold and their identities, accumulates the ratings of all users in the encrypted domain. Then, the encrypted sum is sent to user A along with the encrypted number of similarities above the threshold, L. User A decrypts the sum and L and, computes the average values, obtaining the recommendations. [8]

4. Conclusion

We have presented a highly efficient, privacy-preserving cryptographic protocol for a crucial component of online services: recommender systems. Our construction with a semitrusted third party, the PSP, ensures a protocol where user participation in the heavy cryptographic operations is no longer needed.We also employ data packing to ease the computational and communication burden between the service provider and the PSP. A

cryptographic protocol particularly developed for comparing packed and encrypted values, enables us to compare multiple encrypted data elements in a single operation

5. References

[1] List of Social NetworkingWebsites 2009 [Online]. Available: http://enwikipedia.org/wiki/List_of_social_networking_websites

[2] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.

[3] N. Ramakrishnan, B. J. Keller, B. J. Mirza, A. Y. Grama, and G. Karypis, "Privacy risks in recommender systems," *IEEE Internet Comput.*, vol. 5, no. 6, pp. 54–63, Nov./Dec. 2001.

[4] N. Kroes, Digital agenda, Brussels, May 19, 2011.

[5] R. Agrawal and R. Srikant, "Privacy-preserving data mining," in *Proc. SIGMOD Rec.*, May 2000, vol. 29, pp. 439–450.

[6] Y. Lindell and B. Pinkas, "Privacy preserving data mining," *J. Cryptol.*, pp. 36–54, 2000, Springer-Verlag.

[7] H. Polat and W. Du, "Privacy-preserving collaborative filtering using randomized perturbation techniques.," in *Proc. ICDM*, 2003, pp. 625–628.

[8] H. Polat andW. Du, "SVD-based collaborative filtering with privacy," in *Proc. 2005 ACM Symp. Applied Computing (SAC'05)*, New York, NY, 2005, pp. 791–795, ACM Press.

[9] S. Zhang, J. Ford, and F. Makedon, "Deriving private information from randomly perturbed ratings," in *Proc. Sixth SIAM Int. Conf. Data Mining*, 2006, pp. 59–69.
[10] R. Shokri, P. Pedarsani, G. Theodorakopoulos, and J.-P. Hubaux, "Preserving privacy in collaborative filtering through distributed aggregation of offline profiles," in Proc. Third ACM Conf. Recommender Systems (RecSys'09), New York, NY, 2009, pp. 157–164, ACM.

[11] F.McSherry and I. Mironov, "Differentially private recommender systems:Building privacy into the net," in Proc. 15th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD'09), New York, NY, 2009, pp. 627–636, ACM.

[12] R. Cissée and S. Albayrak, "An agent-based approach for privacypreserving recommender systems," in Proc. 6th Int. Joint Conf. Autonomous Agents and Multiagent Systems (AAMAS'07), New York, NY, 2007, pp. 1–8, ACM.

[13] M.Atallah, M. Bykova, J. Li,K. Frikken, andM. Topkara, "Private collaborative forecasting and benchmarking," in Proc. 2004 ACM Workshop on Privacy in the Electronic Society (WPES'04), New York, NY, 2004, pp. 103–114, ACM.

[14] J. F. Canny, "Collaborative filtering with privacy.," in IEEE Symp. Security and Privacy, 2002, pp. 45–57.

[15] J. F. Canny, "Collaborative filtering with privacy via factor analysis," in SIGIR. New York, NY: ACM Press, 2002, pp. 238–245.

[16] Z. Erkin, M. Beye, T. Veugen, andR. L. Lagendijk, "Privacy enhanced recommender system," in Proc. Thirty-First Symp. Information Theory in the Benelux, Rotterdam, 2010, pp. 35–42.

Authors' Biography

Y.Bhargav had B.Tech from Guru Nanak Engineering College, Ibrahimpatnam, Hyderabad. He is an M.Tech. student in CSE Department of CMR Institute of Technology, Hyderabad. He is currently working for her M.Tech. research project work under the guidance of Mr.P.S.Murthy. His areas of interest include Network Security, Computer Networks, and Programming languages.

P.Sreenivasa Moorthy M.E., (Ph.D) he is currently working as Associate Professor in CSE Department of CMR Institute of Technology, Hyderabad. His areas of interest are Image Processing, DataBases, Networking and Security.

