A SECURE SCHEMA FOR RECOMMENDATION SYSTEMS

Asny P.A[1] and Susanna M. Santhosh [2]

[1] Student, Department of Computer Science and Engineering, MBITS Nellimattom and [2]Assistant Professor, Department of Computer Science and Engineering,MBITS Nellimattom

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. This paper aims to protect the private data against the service provider while preserving the functionality of the system. This paper provides a general framework that, with the help of a preprocessing phase that is independent of the inputs of the users, allows an arbitrary number of users to securely outsource a computation to two non-colluding external servers. This paper use these techniques to implement a secure recommender system based on collaborative filtering that becomes more secure, and significantly more efficient than previously known implementations of such systems.

KEYWORDS

Secure multi-party computation, privacy, recommender systems,secret sharing.

1.INTRODUCTION

Recommendation systems are an important part of the information and e-commerce ecosystems. They represents a powerful method for enabling users to filter through large information and product spaces. Recommendation systems consist of a processor together with multitude of users, where the processor provides recommendations to requesting users, which are deduced from personal ratings that were initially submitted by all the users. Recommender systems based on collaborative filtering method 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 knowledge, or even stolen. Recent studies show that the privacy considerations in online services seems to be one of the most important factors that threaten the healthy growth of the e-business. In a non-cryptographic setup of such a system, the processor is both able to learn all the data submitted by the users and spoof arbitrary, incorrect recommendations. Therefore, it is important to protect the privacy of the users of online services for the benefit of both individuals and business concern.

DOI: 10.5121/ijci.2016.5211

In this work, replaces the recommendation processor by a general two-server processor that satisfies following conditions,

1) The privacy of the ratings and recommendations of the users is maintained.

2) A server that is under adversarial control is unable to disrupt the recommendation process.

In this model the computation is ongoing and outsourced to two external servers that do not collude. This approach allows for the involvement of many users that need only be online for very short time periods in order to provide input data to, or request output data from, the servers. One of the two servers could be the service provider(SP) that wishes to recommend particular services to users, and the other server could be a governmental organisation guarding the privacy protection of users. The role of the second server could also be commercially exploited by a privacy service provider(PSP), supporting service providers in protecting the privacy of the customers.

The major goals of securing the recommendation system is that,

1) Do not want the servers to learn the personal data of users.

2) The correctness of the user outputs is better preserved, because outputs cannot be corrupted by one server on his own.

3) A malicious server might introduce a couple of new dummy users. These dummy users might help him deduce more personal data than is available through the protocol outputs.

In this work used the SPDZ framework,which enables secure multi-party computations[16] in the malicious model, extended it to the client-server model, and worked out a secure recommendation system within this setting. Not only did this lead to a recommendation system that is secure in the malicious model, but also the online phase became very efficient.To extend SPDZ to the client-server model, developed secure protocols that enable users (clients) to upload their datas to the servers, and afterwards obtain the computed outputs from the servers. This required a subprotocol for generating duplicate sharings in the system. To securely compute a recommendation within SPDZ, had to develop secure comparison protocol and secure integer division protocol.

2.RELATED WORKS

Most of the related works on privacy preserving recommendation is secure in the semi-honest model, so parties are assumed to follow rules of the protocol. As mentioned by Lagendijk et al.[1], “Against malicious adversaries achieving security is a hard problem that has not yet been studied widely in the context of privacy-protected signal processing.”

Nikolaenko et al.[5] securely computed collaborative filtering by means of matrix factorization. They used both homomorphic encryption and garbled circuits in a semi-honest security model. In another paper [6], these authors use similar techniques to securely implement the Ridge regression, a different approach of collaborative filtering.

Erkin et al.[2] securely computed recommendations based on collaborative filtering method. They used homomorphic encryption within semi-honest security model just like Bunn[3] and Ostrovsky[4]. Goethals et al stated that although such techniques can be made secure in malicious model, will make them unsuitable for real life applications because of the increased computational and communication costs.

Some schemes with controllable and revocable anonymity provide linkability by adding a tag to a signature. Using the tag associated with a signature, one can check the linkability on the signatures easily and explicitly. For example, a linkable democratic GS scheme is a variant of a democratic GS scheme to support the tag-based link ability. A message-linkable GS scheme was suggested to resist Sybil attacks in the VANET.

In the last several years, a couple of computation protocols have been developed, which are both practical and secure in the malicious model. The idea is to use public-key techniques in a data-independent pre-processing phase, such that cheap information-theoretic primitives can be exploited in the online phase, which makes the online phase efficient. In 2011, Bendlin et al.[7] presented such a framework with a somewhat homomorphic encryption scheme for implementing the pre-processing phase. This has been improved lately by Damgård et al.[8], which has become known as SPDZ (pronounced “Speedz”). Last year, Damgård et al.[9] showed how to further reduce the precomputation effort.

3.PROBLEM DEFINITION

Recommender systems have become an important tool for the personalization of online services. Generating recommendations in online services depends on privacy-sensitive data collected from the user. Traditional data protection mechanisms focus only on access control and secure transmission, which provide security against malicious third parties, but not the service provider. This creates a serious privacy risk for the users.Most of the system is only secure in semi-honest model. So aim to protect the private data against the service provider while preserving the functionality of the systems.Propose amodified version of the standard model for the secure multi-party computation, which is a cryptologic paradigm in which the players jointly perform a single secure computation and then abort.By introducing general two-server processor in such a way that, as long as one of the two servers is not controlled by an adversary and behave correctly.

4.SECURE RECOMMENDATION SYSTEM

4.1.User-Based Collaborative Filtering

Collaborativefiltering(CF) is a popular recommendation algorithm that based its predictions and recommendations on the ratings or behavior of other users in the system. The fundamental assumption behind this method is that other users opinions can be selected and aggregated in such a way that to provide a reasonable prediction of the active user’s preference.To generate recommendations for a particular user in a group of users and items, uses a system based on collaborative filtering, which has the following three steps.

1)Similarities are computed between that particular user and all others.

2) The most similar users are selected by comparing their similarity values with a threshold.

3) The recommendations on all of the items are generated as the average rating of the most similar users.

In collaborative filtering there is one processor R, with N users, and M different predefined items. A small subset of sizeS ($1 \leq S < M$) of these items is assumed to have been rated by each user,reflecting his personal taste. The remaining M – S items have only been rated by a small subset of users that haveexperienced by particular item before. A user that is lookingfor new, unrated items, can ask the processor to produceestimated ratings for theM−S items. The number Nof users can be large, M is in the order ofhundreds, and S usually is a few tens [12].

During initialisation, each user n uploads to processor a list of at most M ratings of items, where each rating V(n,m) is represented by a value within a pre-specified interval. Users can update their rating at any time during the lifetime of the system. A user can, at any time after the initialization, request a recommendation from processor. When the processor receives such a request from a user, computes a recommendation for this user as follows. First, it uses the initial S ratings in each list to determine which other users are considered to be similar to the requesting user, have similarly rated the first S items. The remaining M−S entries in the lists are then used to compute and return a recommendation for the requesting user, consisting of M −S ratings average over all similar users.

To get an idea of required computation we describe the required computational steps. Let Um be the set of users that have rated item m, $S < m \leq M$.

1) Each user uploads his ratings to the processor[1], we assume the first S ratings have been normalized and scaled beforehand. A rating is normalized by dividing it by the length of the vector (V(n,1), . . , V(n,S)), yielding a real number s between 0 and 1. Next, this real number is scaled and rounded to a positive integer consisting of a few bits. The remaining ratings should only be scaled and rounded to an integer with same maximal number of bits.
2) When user A asks for a recommendation, processor computes M − S estimated ratings for A. The similarities SimA,n= $\sum^{s}$ m =1 V(n,m) ・V(A,m) are computedfor each user n.
3) Each similarity value is compared with a public threshold $t \in N+$, and outcome is presented by the bit $\delta n = (t <$SimA,n).
4) The recommendation for user A consists of M – S estimated ratings, the estimated rating for item m, $S < m \leq M$, simply being an average of the ratingsof the similar users: Recm = ($\sum_{n}$∈Um δn ・V(n,m)) ÷ ($\sum_{n}$∈Um δn), where ÷ denotes integer division.
5) The processor send back the recommendation RecS+1 . . . RecM to user A.

4.2.SECURE MODEL

4.2.1.SecretSharing

An external dealer distributes shares of a secret value x to the two servers as follows:

1) The dealer selects a value r uniformly at random.
2) The dealer sends the value r to server 1(SP) and the value $x - r$ to server 2(PSP).

The values x1 = r and x2 = x −r are considered to be the share of SP and the share of PSP, respectively. It should be clear from the description above that the shares x1 and x2 are both individually statistically independent of the secret x, while they together allow to determine the value of x, by adding these shares together.

In addition to the distribution of the shares, the dealer distributes authentication tags on the shares with respect to the authentication code C , defined as C(x, (α,β)) = α ・x + β. Here the value (α, β) is called the authentication key and the value α ・x + β the authenticationtag for the share x.

For every share x1 for SP, the dealer generates a random authentication key (α2, β2), computes the corresponding authentication tag m1 = α2 ・x1 + β2 and sends the key (α2, β2) to PSP, and the share x1 and tag m1 to server SP.

4.2.2.Operations for the Computation Phase

In this describes the operations that are needed for the recommender system. Suppose that servers 1 and 2 hold fixed partial authentication keys $\alpha 1, \alpha 2$.

1) Linear Operations:

Let [*x*] and [*y*] be secret sharings of arbitrary values *x*, *y* and let *c* be a public constants. Show how to non-interactively compute secret sharings for [*x* + *y*], [*cx*] and [*x* + *c*] respectively. An authenticated secret sharing of the sum $z = x + y$ is computed by locally adding the shares, keys, and tags of *x* and *y*. From a sharing [*x*], an authenticated secret sharing of *cx* is computed by local multiplication of the shares, keys, and tags of *x*. To add a public constant to a secret sharing [*x*], one party adds the constant to its share, and the other party adjusts its authentication key.

2) Multiplication:

Let [x] and [y] denotes the secret sharings of arbitrary values x, y, and [a], [b], [c]be a given multiplication triplet such that c = ab. The following sequence of local linear operations and interactions is used to compute a sharing [z] where z = xy, making use of the precomputed multiplication triplets:

- The servers locally compute the secret sharing [v] = [x −a] from [x] and [a], and open it towards each other.

- The servers locally compute the secret sharing [w] = [y −b] from [y] and [b], and open it towards each other.

- The servers locally computes the secret sharing [z] = [xy] = w[a] + v[b] + [c] + vw.

4.2.3.Secure Architecture for Recommendation Systems

This secure framework relies on techniques from the cryptologic area[15] of secure multi-party computation[14]. This model have the following structures. First,inputphasethat use in the secure computation enables the parties to encrypt their respective inputs.Next a computation phase that takes place during which an encrypted output of the function f is computed from the encrypted inputs. Last, an output phasetakes place where the output is decrypted, and then sent to the appropriate parties. Consider secure multi-party computation in the preprocessingmodel,where at some point in time prior to the selection of the inputs, a preprocessing phase takes place that establishes the distribution of an arbitrary amount of correlated data between the parties involved in the computation phase.This data is completely independent of the input data of the parties in the system. The goal of the preprocessing is to remove as much of the complexity and interaction from the actual computation as much as possible, which as a result makes this computation extremely efficient.

Every computation s corresponds with a function *f*, which represented via an arithmetic circuit consisting of basic operations like addition and multiplication. For considering the recommender application, it suffices to consider these basic operations together with the more complex operations of comparison and integer division, which are composed of basic operations.

The outsourcing aspect of the secure computation, the input phase is non-standard in the sense that the inputs are not provided by the two servers. The input phase results in encryptions of the inputs that are suitable for the computation phase of the two-party computation with preprocessing.The figure 1.shows the architecture of a secure recommendation system.
Although the users providing the inputs could in principle take care of the share distribution, these users cannot be trusted to provide the authentication keys and tags, they might be under control of one of the servers. Here introduces an additional structure to the authentication keys in order to enable a secure two-party computation approach.

Once the encryption of the inputs has been established, the computation recursively handles the operations in the circuit while maintaining the encryption structure as constant.Every operation in the circuit is initiated with two encrypted inputs, and produces an encrypted output without leaking any information on the encrypted values provided. So, at the end of the circuit, an encryption of the final output becomes a vailable.

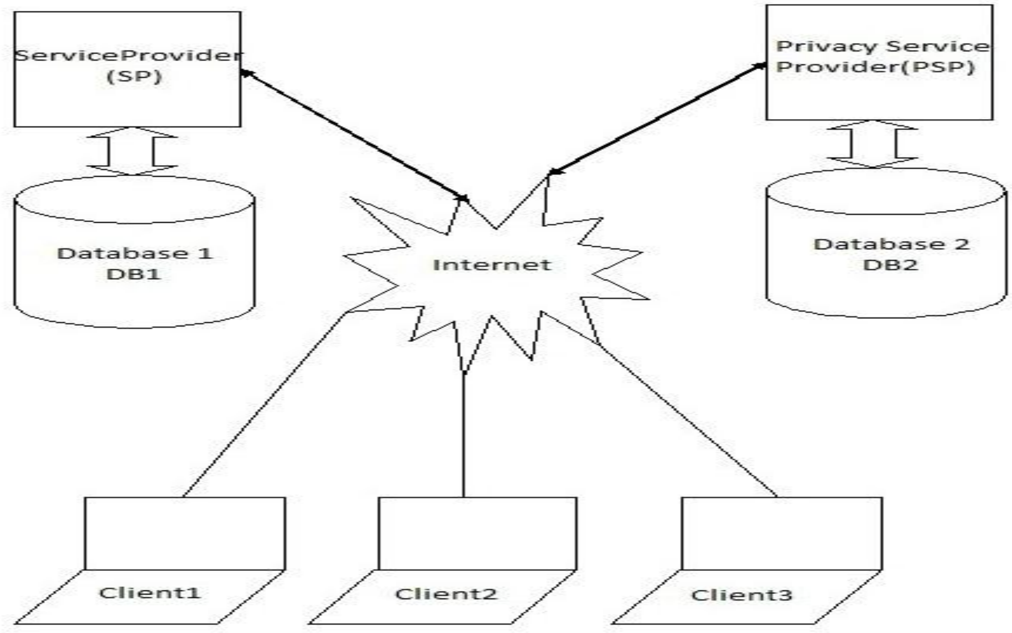

Figure 1. Architecture for A Secure Recommendation systems

The output phase is also non-standard, as the output needs to be revealed to an external parties. Here the idea is that all data related to the encryption of the output is sent back to the relevant users in the system, so that this user can verify the correctness of the shares using the authentication keys and tags provided, and then decrypt the output using the shares.

1)The Input Phase

This framework allows multiple clients to upload any values to the processor,focus on this recommender application[13]. Initially, each user *n* ($1 \leq n \leq N$) will have to upload his ratings *V(n,m)* ($1 \leq m \leq M$) once, before the recommendations can be requested. A user could easily act as a dealer by splitting his rating into two shares, and sending each servers(service provider and privacy service provider) a share accompanied by proper authentication tags.

2)The Output Phase

Although this framework allows the clients to download any value from the processor, focus on our recommender application. After a recommendation has been computed, the outputs Rec*m* $\in(S < m \leq M)$ have to be sent to the requesting users in the system.

5.CONCLUSIONS

In this work provide a highly efficient, privacy-preserving general framework for securing a recommendation system. This framework is then applied to the problem of secure recommendation and, given a sufficient amount of precomputed data, leads to extremely efficient implementations.

While this work focuses on the application of secure recommendation systems, the underlying framework is sufficiently generic for use in other, similar applications, and also easily extends to model variations involving more than two servers.

REFERENCES

[1] Thijs Veugen, Robbert de Haan, Ronald Cramer, and Frank Muller,” A Framework for Secure Computations With Two Non-Colluding Servers and Multiple Clients,Applied to Recommendations”, Ieee Transactions On Information Forensics And Security, Vol. 10, No. 3, March 2015.

[2] R. Lagendijk, Z. Erkin, and M. Barni, “Encrypted signal processing for privacy protection: Conveying the utility of homomorphic encryption and multiparty computation,” *IEEE Signal Process. Mag.*, vol. 30, no. 1,pp. 82–105, Jan. 2013.

[3] Z. Erkin, T. Veugen, T. Toft, and R. L. Lagendijk, “Generating private recommendations efficiently using homomorphic encryption and data packing,” *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 3, pp. 1053–1066, Jun. 2012.

[4] P. Bunn and R. Ostrovsky, “Secure two-party k-means clustering,” in *Proc. 14th ACM Conf. Comput. Commun. Secur.*, 2007, pp. 486–497.

[5] B. Goethals, S. Laur, H. Lipmaa, and T. Mielikäinen, “On private scalar product computation for privacy-preserving data mining,” in *Proc. 7th Int. Conf. Inf. Secur. Cryptol.*, 2004, pp. 104–120.

[6] V. Nikolaenko, S. Ioannidis, U. Weinsberg, M. Joye, N. Taft, and D. Boneh, “Privacy-preserving matrix factorization,” in *Proc. ACMSIGSAC Conf. Comput. Commun. Secur.*, 2013, pp. 801–812.

[7] V. Nikolaenk, U.Weinsberg, S. Ionnidis, M. Joye, D. Boneh, and N. Taft, “Privacy-preserving ridge regression on hundreds of millions of records,” in *Proc. IEEE Symp. Secur. Privacy*, May 2013, pp. 334–348.

[8] R. Bendlin, I. Damgård, C. Orlandi, and S. Zakarias, “Semihomomorphic encryption and multiparty computation,” in *Advances in Cryptology* (Lecture Notes in Computer Science), vol. 6632. Berlin,Germany: Springer-Verlag, 2011, pp. 169–188.

[9] I. Damgård, V. Pastro, N. Smart, and S. Zacharias, “Multiparty computation from somewhat homomorphism encryption,” in *Advances inCryptology* (Lecture Notes in Computer Science), vol. 7417. Berlin, Germany: Springer-Verlag, 2012, pp. 643–662.

[10] I. Damgård, M. Keller, E. Larraia, V. Pastro, P. Scholl, and N. P. Smart, “Practical covertly secure MPC for dishonest majority—Or: Breaking the SPDZ limits,” in *Computer Security* (Lecture Notes in Computer Science), vol. 8134. Berlin, Germany: Springer-Verlag, 2013.

[11] M. Atallah, M. Bykova, J. Li, K. Frikken, and M. Topkara, "Private collaborative forecasting and benchmarking," in *Proc. ACM WorkshopPrivacy Electron. Soc. (WPES)*, 2004, pp. 103–114.

[12] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, *Recommender Systems Handbook*. New York, NY, USA: Springer-Verlag, 2011.

[13] T. Veugen, "Encrypted integer division and secure comparison," *Int J. Appl. Cryptograph.*, vol. 3, no. 2, pp. 166–180, 2014.

[14] BI. Damgård, M. Fitzi, E. Kiltz, J. B. Nielsen, and T. Toft, "Unconditionally secure constant-rounds multi-party computation for equality, comparison, bits and exponentiation," in *Theory of Cryptography*. Berlin, Germany: Springer-Verlag, 2006, pp. 285–304.

[15] R. Canetti, "Universally composable security: A new paradigm for cryptographic protocols," in *Proc. 42nd IEEE Symp. Found. Comput.Sci.*, Oct. 2001, pp. 136–145.

[16] T. Nishide and K. Ohta, "Multiparty computation for interval, equality, and comparison without bit-decomposition protocol," in *PublicKey Cryptography* (Lecture Notes in Computer Science), vol. 4450, T. Okamoto and X. Wang, Eds. Berlin, Germany: Springer-Verlag, 2007, pp. 343–360.

AUTHORS

AsnyP.A. is currently pursuing M.Tech in Computer Science and Engineering in Mar Baselios Institute of Technology and Science,Nellimattom.She completed her B.Tech from Ilahia College of En gineering and Technology,Muvattupuzha. Her specialization in Cyber Security.

Susanna M. Santhosh is currently assistant professor of the Department of Computer Science and EngineeringMar Baselios Institute of Technology and Science,Nellimattom., Kothamangalam, Kerala, India. She received her B-Tech Degree in Computer Science and Engineering in 2010 from College of Engineering, Chengannur and M-Tech from Federal Institute of Science and Technology,Angamaly in 2012. She has around 3 years of teaching experience. Her specializayion in InformationS