On Privacy Preserving Collaborative Filtering: Current Trends, Open Problems and New Issues

Fran Casino*, Constantinos Patsakis‡, Domènec Puig* and Agusti Solanas*†
*UNESCO Chair in Data Privacy
Dept. of Computer Engineering and Mathematics
Universitat Rovira i Virgili
Av.Països Catalans 26, 43007 Tarragona, Catalonia
‡ Distributed Systems Group
School of Computer Engineering and Statistics
Trinity College, Dublin, Ireland
†Corresponding author e-mail: agusti.solanas@urv.cat

***Abstract*—Automatic recommender systems have become a cornerstone of e-commerce, especially after the great welcome of *Web 2.0* based on participation and interaction of Internet users. Collaborative Filtering (CF) is a recommender system that is becoming increasingly relevant for the industry due to the growth of the Internet, which has made it much more difficult to effectively extract useful information. In this paper, we introduce a taxonomy of the different CF families and we discuss the most relevant Privacy Preserving Collaborative Filtering (PPCF) methods in the literature. To understand the inherent challenges of the PPCF, we also conduct an overview of the current tendencies and major drawbacks of this kind of recommender systems, and we propose several strategies to overcome the shortcomings.**

Key words: Privacy Preserving Collaborative Filtering, Electronic Commerce, Recommender Systems.

I. Introduction

Recommender systems [1] derive from knowledge discovery in databases (KDD) [2], [3]. Nowadays, the greatest source of recommendations is the Internet. On the one hand, the Internet provides a wealth of information on a huge variety of products and services that may be useful to potential buyers. On the other hand, such an amount of information may become a problem rather than a solution because it can hinder the decision making. Collaborative Filtering (CF) [4], [5] is a recommender system which aims to make suggestions on items (e.g. books, music, or movies) based on preferences of users that have already acquired and/or rated these items. CF appears in order to provide automatic recommendations in a digital environment. It is actively present throughout the network: (i) E-commerce companies (e.g. Amazon, eBay, Barnes & Noble) take advantage of CF in order to efficiently obtain benefits. (ii) CF is part of the *Web 2.0* concept, which is defined as the new way to use the network. This new trend gives high importance to the active user participation in the infrastructure (e.g. blogs, social networks and information & service portals). (iii) It is used to analyze information preferences. For instance, the websites visited by a user can be monitored and used to recommend similar web pages to other users with similar behaviors. (iv) It is widely used in the musical and audiovisual context, with sites like last.fm, MyStrands, Netflix and Moviefinder that leverage CF to benefit the user in form of advice and also provide a clear market research to the companies.

CF methods use large databases that store information of users on items which have been purchased. These data are modeled as matrices composed of n users and m items and each cell (n_i, m_j) stores user's i evaluation on item j. These stored ratings are represented through a range of values (e.g. between 0 and 10) or simply with a binary vote, which can be positive or negative (or bought and not bought) as in market basket databases. Table I shows a small example of a toy CF matrix. There are many examples of CF referenced databases [6] in the literature: Eachmovie, MovieLens, Jester, and Netflix prize data. These are often used as benchmarks to evaluate the efficiency, quality and robustness of CF methods [7].

TABLE I. Example of data matrix where rows correspond to users and columns correspond to items. Note that Item 3, marked in gray, has not been evaluated yet.

	I_1	I_2	I_3	I_4	I_5	I_6
U_a	2	4			1	2
U_b		3		2		
U_c	3	2			2	2
U_d		5		1		1

The recommendations provided by CF methods are based on the assumption that similar users will be interested in the same products. In this way, items purchased by a user u_a can be recommended to another user u_b, if u_a and u_b are similar. Also, there are approaches that make recommendations based on similarity between items. In this context, many similar users or items perform a neighborhood.

The state-of-the-art proposes a classification of CF methods into three main categories depending on the data they use to make the recommendation: memory-based methods (which use the full matrix with all the ratings), model-based (which use statistical models and functions of the data matrix but not the complete data matrix) and hybrid methods (which combine the previous methods with content-based [6] recommendation methods). In memory-based CF, recommendations are made in two steps: neighborhood search and recommendation prediction. Given a user u_a, correlation and distance functions are used to compute its neighborhood. The most used correlation

functions are Pearson Correlation [5], Cosine similarity [8] and the Euclidean distance. The similarity between users can also be computed in a much more efficient way, according to the behavior when they vote. Examples of these are shown in [9], in which the tendencies of users are calculated, or in [10], in which the concordances between users are computed with privacy. In general, to compute similarities between users, there should be a sufficient number of commonly rated items. Once we have formed the neighborhood of the user u_a, we can make a recommendation using the methods proposed, for instance, in [11] and [5]. These methods can be used to predict a vote or to recommend top-N items for u_a.

Model-based CF methods create a model from the full matrix on which to make recommendations. The emergence of these methods is justified by the restrictions of the memory-based CF in terms of scalability, complexity of calculation and sparseness. Some well-known methods to reduce the dimensionality of a matrix are Singular Value Decomposition (SVD)[1] and Principal Component Analysis (PCA)[2]. However, the use of dimensionality reduction methods could affect the quality of the recommendations since they reduce the data range. There exists a huge variety of model-based CF methods: dimensionality reduction methods (SVD [2], RSVD, Improved RSVD, NSVD2 and SVD++ [9], [12]), linear regression methods [13], clustering methods [6], [14], and Bayesian network models amongst others [6].

Hybrid CF methods combine memory-based and model-based methods in a way to preserve the advantages of the algorithms involved whilst minimizing drawbacks and deficiencies. Examples of these are the Personality Diagnosis [6] and the Probabilistic memory-based model [6]. Hybrid methods can also be obtained by the combination of memory-based and model-based methods with content-based recommender systems. Some examples [6] are: Filterbots, Fab and Ripper.

Regardless of the CF method used, there are various limitations inherent to this kind of recommender system. Some of the most important limitations [6], [15] are *sparseness*, *scalability*, *cold start*, *shilling*, *synonymy*, *bribing*, *copy-profile attacks*, and the lack of privacy.

The use of trust systems has increased on the Internet in the last years. A trust statement is defined as the explicit opinion expressed by a user on another user regarding the perceived quality of certain characteristics of this user [15]. The trust concept is widely used, for instance, in search engines such as Google, which uses global trust metrics [16] and, also, in e-commerce (ebay.com) where users express their level of satisfaction after purchasing a product.

Trust statements expressed by every user can be aggregated to produce a community or a neighborhood [17] as seen, for instance, in social networks. Since the ratings scored by users are sparse, often finding similar users becomes impossible. By the application of a trust-based heuristic in trust networks, we can find the users which are most trustworthy for an active user. We can combine information provided by trust networks with CF matrices to form a Trust-Aware recommender system [15], which deals better with problems like data sparsity, cold start users and fake identities. Despite the benefits, trust systems also have problems like *controversial users*, that can be minimised by using local trust metrics [16].

The contribution of this paper is twofold: it firstly presents a concise state-of-the-art of the existing CF methods and a classification of the most relevant Privacy Preserving Collaborative Filtering (PPCF) methods. Secondly, it discusses current drawbacks of these methods and proposes new improvements. The rest of the paper is organised as follows. Section II introduces the current trends on PPCF and classifies the most relevant methods in the literature. Section III discusses open problems of PPCF and devises new issues and strategies for the design of CF systems. Finally, section IV concludes the paper and provides directions for future research.

II. Privacy Preserving Collaborative Filtering

As previously stated, the widespread use of CF on the Internet provides great opportunities and benefits to both companies and users, but there is a major drawback: the lack of users' privacy. The importance of privacy in CF systems is emphasised by the growing pace at which information of each user is collected and stored. Careless management of personal information, besides being illegal, could lead to serious consequences for both users, whose information is stored, as well as businesses.

One of the main problems in CF is that customers who believe that their preferences/profiles may be exposed, do not give their assessment on a particular item or, if given, they do it incorrectly or distorted [18]. This user behavior, derived from the lack of privacy feeling, incurs in a reduction of both the number of assessments as well as their quality. Another drawback is that companies can acquire data of the preferences of many users in a given market, getting a big advantage over new competitors if they decide to expand into other markets. Another user's disadvantage is the existence of large monopolies on the Internet (Google, Amazon) so their data could be transferred between different entities, which are managed by large companies, without the user's awareness.

Interestingly, although CF methods with privacy obfuscate and/or hide information of user profiles, the creation of communities of similar users, which is a very common fact in the network, can become a double-edged sword. First, users can easily find reliable recommendations on items from communities in a particular context. Second, it may incur a *homophily* problem in the network, more specifically a problem of *value homophily* [19], so that recommendations outside the context of the community would give results with little sense, precisely because of the homogeneity of the group. In order to solve the privacy issues raised by the systematic collection of private information, which is required for the proper use of CF, current research focuses on Privacy Preserving Collaborative Filtering (PPCF) methods.

In a dynamic market like the Internet, companies may be interested in cooperating to obtain better recommendations for their customers. Due to privacy and business concerns, data should not be disclosed between companies. In this context,

[1]Method that uses a factorization matrix of the type $M = U\Sigma V^*$ where M is an mxn matrix, U is an unitary mxm matrix, Σ is an mxn matrix with non negative values in its diagonal, and V^* is an unitary nxn matrix.

[2]Method that computes a projection of the data matrix, in which the data remains as best represented as possible, in order to reduce its dimensionality.

data might be partitioned between various parties in different ways:

Vertical partitioning (VP) in which companies own disjoint sets of items but with the same users.

Horizontal partitioning (HP) in which different parties hold disjoint sets of users with opinions of the same items.

Arbitrary partitioning (AP) in which there is no pattern of how data are distributed. If the entire set is defined by an $m \times n$ user-item matrix, one party A holds a subset of users $m_a \leq m$ whilst another party B holds the rest $m_b = m - m_a$ and the same is applied for items. Notice that VP and HP are specific cases of AP. This is the most realistic scheme in e-commerce, where companies offer different products within the same context and users can be interested in an undetermined number of different companies.

As shown in the literature, there are several ways to protect privacy in databases. To anonymise profiles in large databases we can use methods to provide k-anonymity [20]. For its interesting properties, public key homomorphic cryptosystems [21], secure multi-party computations [23] and cryptographic protocols are used on the Internet. Other methods add noise to the data to distort their values, in a way that affects the least possible the statistical features of the matrix, such as the average ratings of the users and the articles.

In the next subsections, we propose a classification of the PPCF approaches according to the location of the information into centralised methods and decentralised methods, and we summarise the most relevant ones in each category.

A. Centralised PPCF Methods

A PPCF method is centralised if it uses a third party to make intermediate calculations amongst users or entities. A method is also considered centralised if the ratings are stored in a single server where recommendations and predictions are computed. Cases in which data are partitioned are not considered centralised, as the data are distributed amongst different parties.

Several proposals of centralised PPCF methods can be found in the literature. A clear example is the method proposed by Polat et al. in [24], which shows a technique to achieve a good balance between privacy and quality of recommendations. Due to privacy concerns, the central server should not store real users' data. To avoid the disclosure of the real users' profiles, users distort her data by adding random vectors, which are created following a Gaussian distribution, before sending them to the server. To obfuscate the data, the server decides a range $[-x, x]$, known by users, to truncate the random values generated by the Gaussian distribution. Later, each u_i calculates the z-scores[3] z_{ij} of the elements that u_i has rated. Finally, each u_i creates n_i uniform random numbers r_{ij} in the range $[-x, x]$, where n_i is the number of rated items by the user. After that, each u_i aggregates the random numbers to the z-score ratings and generates the disguised z-scores z'_{ij} = z_{ij}+ r_{ij}. Subsequently, users send the disguised z-scores to the server. After obtaining the disguised z-scores z'_{ij} of various users, the server is able to sent the aggregated information which the active user needs to compute the predictions locally.

In a more recent work, to improve the calculation speed and the efficiency showed in [24], Polat et al. proposed a method in [25] that uses SVD to reduce the dimensionality of the original matrix. Its objective is to make predictions using a new matrix with improved characteristics such as reduced data dispersion.

In [26], Zhang et al. propose an item-based PPCF scheme which adds noise to the ratings in order to protect the users' privacy. In this case, perturbations are made by their importance in the recommendation process instead of the classic item-invariant perturbation already used, for instance, in [24], [25]. Their results show that this kind of perturbation results in a privacy gain without a lower prediction quality respect of the item-invariant perturbation.

Following the line of data obfuscation, we find the method *Nearest Neighbor Data Substitution* (NeNDS) proposed by Parameswaran et al. [27]. This proposal uses a CF server where different sources combine their data to obtain enough information to better deal with data sparsity. The scheme assumes that entities have three types of databases: User-info, Item-info and Ratings-info. In this case, databases are obfuscated and sent to the central server, which performs a data aggregation and returns the databases with new values to each source. In order to preserve the properties of each database content, each field is treated separately and permutations of similar items are performed to obfuscate the data. The benefit is an obfuscated database without affecting severally its subgroups values. The previously obfuscation could be weak in terms of privacy. For this reason, once the obfuscated database is obtained, the authors use geometric transformations (scaling, rotation and translation) because they preserve both the database subgroups and the distances between elements.

B. Decentralised PPCF Methods

All those methods which use the members of a distributed network, in order to perform intermediate calculations and predictions on items ratings, can be classified as decentralised PPCF methods. These members will be considered in most cases as users. The use of decentralised schemes generally ensures that the information exposed is much less than in the case of centralised systems, but this entails the use of expensive protocols and more complex calculations. Typically, in decentralised PPCF methods, users store their own ratings. Well-known PPCF with partitioned data schemes, which involve different parties sharing their data to perform CF with more referrals, are also considered decentralised methods.

Several approaches with partitioned market basket databases have been proposed in the literature [28]–[30]. These kind of databases are suitable to make top-N recommendations with high accuracy and low computational cost due to its binary ratings contents. Despite of that, the vast majority of the state-of-the-art focuses in PPCF with numerical rating contexts.

In [31], Polat et al. propose an item-based PPCF method with APD, which has offline and online stages to increase the overall efficiency. The offline stage implies the model construction, which is the creation of the item-based neighborhoods.

[3]Transforming the original distribution in one which the mean equals to 0 and the standard deviation is equal to 1.

Privacy preserving protocols, which use data perturbation and homomorphic encryptions, are used to protect users' data. These protocols are utilised to compute and exchange the necessary information amongst the parties to construct the model. Once the model is created, online recommendations can be performed using the active user u_a data and the constructed model.

In [32], Yakut et al. propose the use of SVD to conduct PPCF on both vertically and horizontally partitioned data. When data are partitioned between two companies P and Q which have R_P and R_Q matrices respectively, the authors need to handle four principal issues in order to conduct the SVD of the full matrix, defined as $R = USV^T$. These issues are to remove the sparsity of R, normalise the ratings, compute $R^T R$ and, finally, find the column vectors of U. They use privacy-preserving protocols to calculate and securely exchange all the data involved. Finally, both parties have the estimated U_p, S_p, and V_p matrices, and can then construct the SVD model to make predictions or to recommend top-N items to a user.

There are also privacy preserving approaches with Trust-Aware systems as in [33]. Here, Dokoohaki et al. express the problem of disclosing trust network data because this information may reveal the users' behaviors and communities. To avoid that, they present a scheme which is considered decentralised because trust computations are made locally by each user.

Privacy preserving Trust-Aware systems have also been proposed with VPD. The scheme proposed in [34] is composed of both an offline and an online stage. First of all, an offline approach is used to calculate partial trusts between users with privacy. After these values are computed, a distance-based private sorting algorithm (DPSA) is utilised by the parties to determine the neighborhood of each user. The neighborhood will be formed by her k most trusted users. During the offline process, parties use DPSA protocol n times to determine neighbors for all n users. Once the neighborhoods are calculated, they use a private recommendation protocol which uses Paillier's [21] homomorphic encryption, in order to provide online predictions.

In [35], Hsieh et al. use the well-known Pearson Correlation method to compute the similarities between users on HPD. To compute the Pearson Correlation in HPD, the preference of the user u_i over item i_j is disclosed because the mean value of u_i needs to be shared amongst the entities. To avoid that disclosure, authors propose a secure protocol to compute the Pearson Correlation between users using ElGamal homomorphic encryption. Once the neighborhoods are established, recommendations can be computed straightforward. In a later approach, as the numerator of Pearson Correlation is a scalar product, Zhan et al. [36] use a secure protocol that calculates the scalar product between two vectors $\vec{X}_a$ and $\vec{X}_b$.

In [37], J. Canny proposes a peer-to-peer recommendation scheme using encrypted data. In this scheme, users have their own rating vector and only the aggregates are exposed. Users can decide whether to share their data for CF or not, so this scheme may encourage the formation of communities rather than the use of similarities to compute recommendations. To obtain the ratings' set with which to make recommendations, the author uses an iterative SVD method, which is computed using the conjugate gradient method[4] [38]. Later, in [39], Canny proposed the use of Expectation Maximization factor analysis [40], which deals better with sparse data, to compute the reduced dimensionality matrix. In this case, the author shows a gain in speed and a good quality of recommendations with respect to [37].

Berkvosky et al. [41] also propose the use of a peer-to-peer decentralised scheme to perform PPCF. In this case, each user has its own rating vector and there is no need for a central server or trust party. In the case that an active user u_a sends a request over an item rating i_a, each user decides whether to provide her data for CF or not. If the user u_b decides to collaborate, she calculates the similarity between u_b and u_a using the Cosine similarity metric [6] and sends the result to u_a, with her valuation of the item on which the request was made. Once the data are collected, the k nearest neighbors of u_a are selected to determine the prediction value of i_a for u_a, and a weighted average is calculated according to their proximity.

In [42], Tada et al. propose a PPCF method based on similarity between items. They argue that similarities between items can be made public without privacy leaks because users' profiles are not exposed, neither the similarities between them. Authors use public key Paillier cryptosystem [21] for its homomorphic properties and its faster decryption process (compared with ElGamal), to calculate the simple Cosine correlation between items [6] with privacy. In [43], Kikuchi et al. propose a scheme which is similar to the one already showed in [42]. Here, to reduce the overhead of the proposed protocols, the authors perform item clusters and user clusters. In both cases, the results show a reduction in both communication and computational costs with respect to the results obtained with the original data set.

Basu et al. [44] propose the use of cloud computing to perform weighted Slope One based PPCF, which uses item deviations to make predictions. In this scheme, users store their own values. To decouple the users from their ratings when they are submitted, the authors propose the use of well-known anonymization systems. The proposed scheme has two differentiated parts. In the first part, the cardinalities and the deviations between items are computed and stored in a cardinality matrix and a deviation matrix, respectively. Since users are not related with the information they provide and any concrete rating is sent, these operations do not disclose information. Once the above matrices are obtained, users must send their rating vector to the cloud in order to make predictions. To prevent the disclosure of data, their profiles are sent encrypted using a modified version of the public key Paillier [21] cryptosystem. This modification is needed to operate with negative numbers because deviations, which are involved in the prediction process, may not be positive. The cloud receives the user profile encrypted with the public key, and conducts homomorphic operations with the data it has and the user's profile information to obtain the encrypted recommendations. Later, it sends the encrypted recommendations back to the user, who will decrypt them with her private key.

[4]The conjugate gradient method is an iterative algorithm that solves lineal equation systems which matrices are symmetric[5] and positive defined.

[5]A matrix M is symmetric if M is an squared matrix and its equal to its transposed matrix M^T.

III. Current Trends, Open Problems and New Issues

Current tendencies to achieve better accuracy on recommendations focuses on how to obtain more information. The approaches that use trust networks [15] made on partitioned databases, and the extra information provided by content-based methods, help to deal with cold users and sparseness problems. In that sense, well-known imputation methods [40] could also be used to fill CF matrices and mitigate the problems due to the lack of information. According to a previous study of the dataset, the most fitted imputation method can be selected to obtain accurate and realistic results.

Large multivariate datasets have several well-known problems. We consider the proper detection of outlier users (i.e. *outliers*) as a significant issue. In this way, several approaches to deal with outliers in large multivariate datasets have been proposed [45], [46]. The presence of outliers in datasets affect the quality of the data and, therefore, the applied CF algorithms may return biased results due to its presence. For instance, clustering algorithms may incorrectly select the proper users in order to form a group or a cluster may be wrongly divided into small pieces.

In the privacy area, the current trend is to use secure computation protocols and public key homomorphic properties to protect the users' data whilst CF is performed. Although the majority of these approaches are optimized in order to minimize the time issues and, also, to overcome the dynamic addition of information to the dataset by minimizing the cost of upgrading the algorithms' results, most of the state-of-the-art proposed schemes are not suitable to satisfy the real-time requirements. For this reason, besides accuracy and privacy, as CF methods are widely used to recommend products to users in real time, computation and communication overheads also need to be considered. This fact adds the variable *time* in the well-known accuracy vs. privacy trade-off problem.

Several approaches [44] propose to decouple the users from their ratings using recognised techniques such as mix-servers or anonymization systems. Although they are not methods which guarantee security on their own, their use should be generalised to enhace the privacy of already known CF & PPCF methods.

As previously stated, it is possible to classify PPCF in centralised and decentralised methods. Centralised schemes offer several benefits but also privacy issues. On the one hand, the overall efficiency is increased because extra communication costs between users and third parties are avoided. On the other hand, in centralised methods, data are managed by one party which has total control over it, with the privacy issues implied if the data have a low protection level. No one can guarantee that companies sell information to other parties due to bankrupcity [37]. However, we can use SDC well-known methods [47]–[49] to provide k-anonymity [20] and also public key criptograhpic systems [21] to store that sensitive and private information in a secure way and therefore prevent its disclosure.

Decentralised schemes usually have a higher data protection than centralised. This happens because involved users and parties manage their own datasets, and the data sent between them are usually controlled by secure protocols, in order to avoid illegal disclosures of private data which could also give an economic benefit to competitors. That extra privacy gain incurs in an expensive communication overhead that could affect real time recommendations.

Although time issues can be partially addressed by incremental upgradings of the ratings matrix in a centralised scheme, this solution is difficult to be applied on schemes where users manage their own ratings and any third party is used to perform intermediate calculations, because in that case, any data are stored. In addition, users who control their own profiles tend to form communities with already known users or with those having similar behaviors. As already commented in Section I, this fact may incur in a well-known homophily problem [19], which could be partially solved by the addition of different communities, in order to create a large clustered-society by means of clustering techniques. In this way, users seeking for quality recommendations on a given context could have enough information.

IV. Conclusions and Future Work

Collaborative Filtering is a recommender system that allows automatic recommendations to sets of users in lots of contexts. In this paper, we conducted a brief overview of the state-of-the-art of the different CF families and methods. Despite the great advantages of CF, we have highlighted the significant risks related to the users' privacy. Therefore, we performed an analysis of the most important PPCF methods and we differentiated them mainly by whether they use a centralised or distributed architecture to protect the data and make recommendations. A brief analysis of the major drawbacks of PPCF has also been conducted and several ways to overcome these problems have been discussed.

Although a large amount of CF methods have been proposed, a more in-deep study is needed since there are still challenges to overcome. The adequate protection of the privacy of users, whilst keeping a high degree of accuracy and quality of the recommendations, is probably one of the most significant challenges. We consider that future research should be focused on how to better deal with sparse data whilst efficiently protecting the privacy of users, all done with the lowest possible communication and computation overhead.

Acknowledgments and disclaimer

This work was partly supported by the Government of Catalonia under grant 2009 SGR 1135, by the Spanish Government through projects TIN2011-27076-C03-01 "CO-PRIVACY" and CONSOLIDER INGENIO 2010 CSD2007-00004 "ARES", by the European Comission under FP7 projects "DwB" and "Inter-Trust", and by La Caixa under program RECERCAIXA 2012. The authors are with the UNESCO Chair in Data Privacy, but they are solely responsible for the views expressed in this paper, which do not necessarily reflect the position of UNESCO nor commit that organisation.

References

[1] P. Resnick and H. Varian, "Recommender systems," *Communications of the ACM*, vol. 40, no. 3, pp. 56–58, 1997.

[2] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Application of dimensionality reduction in recommender system-a case study," *ACM WebKDD 2000 Web Mining for ECommerce Workshop*, vol. 1625, no. 1, pp. 264–8, 2000.

[3] U. Fayyad and G. Piatetsky-Shapiro, "Advances in knowledge discovery and data mining," *AAAI/MIT Press*, 1996.

[4] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, vol. 35, no. 12, pp. 61–70, 1992.

[5] P. Resnick, N. Iacovou, and M. Suchak, "GroupLens: an open architecture for collaborative filtering of netnews," *Proceedings of the ACM conference on Computer supported cooperative work CSCW*, vol. pp, no. 3, pp. 175–186, 1994.

[6] X. Su and T. M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," *Advances in Artificial Intelligence*, vol. 2009, no. Section 3, pp. 1–19, 2009.

[7] J. L. Herlocker, J. a. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 5–53, Jan. 2004.

[8] J. Breese, D. Heckerman, and C. Kadie, "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," *UAI 98*, pp. 43–52, 1998.

[9] F. Cacheda, V. Carneiro, D. Fernández, and V. Formoso, "Comparison of collaborative filtering algorithms," *ACM Transactions on the Web*, vol. 5, no. 1, pp. 1–33, Feb. 2011.

[10] N. Lathia, S. Hailes, and L. Capra, "Private Distributed Collaborative Filtering Using Estimated Concordance Measures," *Proceedings of the 2007 ACM conference on Recommender systems RecSys 07*, p. 1, 2007.

[11] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms," *Proceedings of the 10th international conference on World Wide Web*, vol. 1581133480, no. 15, pp. 285–295, 2001.

[12] Y. Koren and R. Bell, "Advances in collaborative filtering," *Recommender Systems Handbook*, pp. 43–52, 2011.

[13] D. Lemire and A. Maclachlan, "Slope one predictors for online rating-based collaborative filtering," *Society for Industrial Mathematics*, vol. 05, no. 12, pp. 471–475, 2005.

[14] S. Lloyd, "Least squares quantization in PCM," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[15] P. Massa and P. Avesani, "Trust-aware collaborative filtering for recommender systems," *On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE, 492-508.*, vol. 3290, no. 8, pp. 492–508, 2004.

[16] ——, "Trust metrics on controversial users: balancing between tyranny of the majority," *International Journal on Semantic Web and Information Systems*, pp. 1–21, 2007.

[17] J. Golbeck, "FilmTrust: Movie Recommendations from Semantic Web-based Social Networks," in *ISWC2005 Posters Demostrations*, 2006, pp. 1314–1315.

[18] L. Cranor, J. Reagle, and M. Ackerman, "Beyond concern: Understanding net users' attitudes about online privacy," The Internet Upheaval: Raising Questions, Seeking Answers in Communications Policy, Tech. Rep., 2000.

[19] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a Feather: Homophily in Social Networks," *Annual Review of Sociology*, vol. 27, no. 1, pp. 415–444, 2001.

[20] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems*, vol. 10, no. 5, pp. 557–570, 2002.

[21] P. Paillier, "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes," *Advances in Cryptology EUROCRYPT 99*, vol. 1592, pp. 223–238, 1999.

[22] A. C. Yao, "Protocols for secure computations," *23rd Annual Symposium on Foundations of Computer Science (sfcs 1982)*, pp. 160–164, Nov. 1982.

[23] H. Polat, "Privacy-preserving collaborative filtering using randomized perturbation techniques," *Third IEEE International Conference on Data Mining*, pp. 625–628, 2003.

[24] H. Polat and L. Hall, "SVD-based Collaborative Filtering with Privacy," *Proceedings of the 2005 ACM symposium on Applied computing SAC 05*, pp. 791–795, 2005.

[25] S. Zhang, J. Ford, and F. Makedon, "A privacy-preserving collaborative filtering scheme with two-way communication," *Proceedings of the 7th ACM conference on Electronic commerce - EC '06*, pp. 316–323, 2006.

[26] R. Parameswaran and D. M. Blough, "Privacy Preserving Collaborative Filtering Using Data Obfuscation," *2007 IEEE International Conference on Granular Computing (GRC 2007)*, pp. 380–380, Nov. 2007.

[27] H. Polat and W. Du, "Privacy-preserving top-N recommendation on distributed data," *Journal of the American Society for Information Science*, vol. 59, no. 7, pp. 1093–1108, 2008.

[28] C. Kaleli and H. Polat, "P2P collaborative filtering with privacy," *Turkish Journal of Electric Electrical Engineering and Compute Science*, vol. 18, no. 1, pp. 101–116, 2010.

[29] I. Yakut and H. Polat, "Estimating NBC-based recommendations on arbitrarily partitioned data with privacy," *Knowledge-Based Systems*, vol. 36, pp. 353–362, Dec. 2012.

[30] ——, "Arbitrarily distributed data-based recommendations with privacy," *Data & Knowledge Engineering*, vol. 72, pp. 239–256, Feb. 2012.

[31] ——, "Privacy-Preserving Svd-Based Collaborative Filtering on Partitioned Data," *International Journal of Information Technology & Decision Making*, vol. 09, no. 03, pp. 473–502, May 2010.

[32] N. Dokoohaki, C. Kaleli, H. Polat, and M. Matskin, "Achieving Optimal Privacy in Trust-Aware Social Recommender Systems," *Lecture Notes in Computer Science*, vol. 6430, no. Social Informatics, pp. 62–79, 2010.

[33] C. Kaleli and H. Polat, "Privacy-Preserving Trust-Based Recommendations on Vertically Distributed Data," pp. 376–379, 2011.

[34] C.-L. Hsieh, J. Zhan, D. Zeng, and F. Wang, "Preserving Privacy in Joining Recommender Systems," pp. 561–566, 2008.

[35] J. Zhan, I. Wang, and C. Hsieh, "Towards efficient privacy-preserving collaborative recommender systems," *GrC 2008.*, pp. 778–783, 2008.

[36] J. Canny, "Collaborative filtering with privacy," *Security and Privacy, 2002. Proceedings. 2002 IEEE*, pp. 45–57, 2002.

[37] E. Polak., *Computational methods in optimization: a unified approach.*, New York, USA, 1971.

[38] J. Canny, "Collaborative Filtering with Privacy via Factor Analysis," *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, vol. 14, pp. 238—-245, 2002.

[39] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society Series BMethodological*, vol. 39, no. 1, pp. 1–38, 1977.

[40] S. Berkvosky, F. Ricci, Y. Eytani, and T. Kuflik, "Enhancing Privacy and Preserving Accuracy of a Distributed Collaborative Filtering," *Proceedings of the 2007 ACM conference on Recommender systems RecSys 07*, pp. 9–16, 2007.

[41] M. Tada, H. Kikuchi, and S. Puntheeranurak, "Privacy-Preserving Collaborative Filtering Protocol Based on Similarity between Items," pp. 573–578, 2010.

[42] H. Kikuchi, H. Kizawa, and M. Tada, "Privacy-Preserving Collaborative Filtering Schemes," pp. 911–916, 2009.

[43] A. Basu, J. Vaidya, H. Kikuchi, and T. Dimitrakos, "Privacy-preserving Collaborative Filtering for the Cloud," pp. 223–230, 2011.

[44] S. Ramaswamy, R. Rastogi, and K. Shim, "Efficient algorithms for mining outliers from large data sets," *ACM SIGMOD Record*, pp. 1–20, 2000.

[45] P. Das and D. Mandal, "Statistical Outlier Detection in Large Multivariate Datasets," *acsu.buffalo.edu*, no. x, pp. 1–9.

[46] A. Solanas, A. Martínez-Ballesté, and U. González-Nicalás, "A variable-MDAV-based partitioning strategy to continuous multivariate microaggregation with genetic algorithms," in *International Joint Conference on Neural Networks(IJCNN)*, 2010, pp. 1–7.

[47] A. Solanas and A. Martínez-Ballesté, "V-MDAV: Variable group size multivariate microaggregation," in *COMPSTAT 2006*, 2006, pp. 917–925.

[48] A. Solanas and R. D. Pietro, "A linear-time multivariate micro-aggregation for privacy protection in uniform very large data sets," in *MDAI*, 2008, pp. 203–214.