A k-Anonymous approach to Privacy Preserving Collaborative Filtering

Fran Casino[a], Josep Domingo-Ferrer[b], Constantinos Patsakis[c], Domènec Puig[b], Agusti Solanas[a,]

[a]Smart Health Research Group. Dept. of Computer Engineering and Mathematics, Universitat Rovira i Virgili, Av. Països Catalans 26, 43007 Tarragona, Catalonia. Spain
[b]UNESCO Chair in Data Privacy. Dept. of Computer Engineering and Mathematics, Universitat Rovira i Virgili, Av. Països Catalans 26, 43007 Tarragona, Catalonia. Spain
[c]Distributed Systems Group, School of Computer Engineering and Statistics, Trinity College, Dublin, Ireland

Abstract

This article proposes a new technique for Privacy Preserving Collaborative Filtering (PPCF) based on microaggregation, which provides accurate recommendations estimated from perturbed data whilst guaranteeing user k-anonymity. The experimental results presented in this article show the effectiveness of the proposed technique in protecting users' privacy without compromising the quality of the recommendations. In this sense, the proposed approach perturbs data in a much more efficient way than other well-known methods such as Gaussian Noise Addition (GNA).

Keywords: Privacy Preserving Collaborative Filtering, Microaggregation, Electronic Commerce, Recommender Systems

1. Introduction

Recommender systems [1] have become a fundamental mechanism to provide users with useful selected information, which could be effective to optimise a large amount of decisions, such as product purchase, films selection, etc. Especially, from the great growth of Internet users, the large amount and variety of available information becomes a problem since it is difficult

[*]Corresponding author
Email address: agusti.solanas@urv.cat (Agusti Solanas)

for users to determine the suitable information to optimise their process of decision making. Taking into account this context, the recommendations based on the Internet are especially relevant for certain types of industry, such as e-commerce. Thus, the Internet provides a wealth of information on a huge variety of products and services that may be useful to potential buyers. However, this wealth of information may become a problem rather than a solution because it can hinder the decision making. Recommender systems are a useful alternative to search algorithms since they help users discover items they might not have found by themselves. Interestingly enough, recommender systems are often implemented using search engines indexing non-traditional data, this being attained through two types of strategies, namely Collaborative Filtering and Content-Based Filtering [2].

Collaborative Filtering (CF) [3, 4] is a recommender system which was introduced to provide automated recommendations in a digital environment. CF is particularly relevant in e-commerce to make suggestions on items (*e.g.* books, music, or restaurants) based on users' preferences that have already acquired and/or rated these items. The underlying idea of CF is that a user will prefer items that like-minded users prefer. Therefore, the recommendations provided by CF methods are based on the assumption that similar users, in the sense of similar interests or behaviours, will be interested in the same products. Thus, items purchased by a user U_a can be recommended to another user U_b, if U_a and U_b have similar interests or similar behaviours.

Currently, CF systems are involved throughout a wide variety of applications. For instance in e-commerce (*e.g.* Amazon, Barnes & Noble) to recommend similar products or in search engines (*e.g.* Google) to recommend similar sites to users with similar interests. Many multimedia sites (*e.g.* last.fm, MyStrands, Netflix, or Moviefinder) make use of CF to propose relevant content. The adoption of CF provides these applications with two main benefits. Firstly, their users receive good advices, improving user experience and the quality of the service. Secondly, CF provides a clear market overview to the companies by exploiting the *Web 2.0* concept; the new way to use the Internet, by giving high relevance to active user participation in the infrastructure (*e.g.* blogs, social networks, or information & service portals).

In order to make predictions, CF methods use large databases that store information regarding the relationships between sets of users and items. These data are modelled as matrices composed of n users and m items, and each cell (i, j) stores the evaluation of user i on item j. Therefore, a value is assigned, which can be within a range of values (*e.g.* between 0 and 10) or simply with binary votes (positive/negative, or bought/not bought) as in market basket databases. There are many examples of CF referenced databases in the literature, such as Eachmovie, MovieLens, Jester, or Netflix prize data. These databases are frequently used as benchmarks to evaluate the efficiency, quality and robustness of CF methods [5].

CF methods can be classified into three main categories according to the data they use to make the recommendation [6]: memory-based methods, which use the full matrix with all ratings; model-based methods, which use statistical models and functions of the data matrix but not the complete data matrix; and hybrid methods, which combine the two previous strategies with content-based recommendation methods.

In memory-based CF, recommendations are made in two steps: (i) neighbourhood search and (ii) recommendation prediction. Given a user U_a, correlation and distance functions are used to compute his neighbourhood. The most common correlation and distance functions used are the Pearson correlation, the cosine similarity and the Euclidean distance. The similarity between users can also be computed in a much more efficient way, according to their behaviour when they vote, for example, by using tendencies [7]. Once the neighbourhood of U_a is determined, recommendations can be computed using, for instance, the methods described in [4, 8]. These methods can be utilised to predict a vote or recommend the 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 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) and Principal Component Analysis (PCA). 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, RSVD, Improved RSVD, NSVD2 and SVD++), linear regression methods [9], clustering methods [10], and Bayesian network models amongst others [6].

Hybrid CF methods combine memory-based and model-based methods, in such a way to preserve the advantages of the algorithms involved and neutralise their shortcomings. Examples of these methods are the Personality Diagnosis [11] and the Probabilistic memory-based model [12].

Regardless of the CF method, there are several limitations inherent to this kind of recommender systems. Some of the most important limitations [6, 13, 14] are *sparseness*, *scalability*, *cold start*, *shilling*, *synonymy*, *bribing*, *copy-profile attacks*, and the lack of privacy.

1.1. Contribution and Plan of the Article

Privacy is a fundamental right and privacy protection is a hot topic to which many research efforts have been devoted from a variety of fields [15, 16, 17, 18]. Thus, amongst all the aforementioned open problems of CF, in this article we concentrate on the protection of the privacy of users involved in CF processes. We perform a classification of PPCF state-of-the-art methods in centralised and decentralised approaches and we introduce Statistical Disclosure Control and microaggregation concepts in order to be used in the CF context. We recall and analyse a novel method for Privacy Preserving Collaborative Filtering based on microaggregation that we briefly introduced in [19]. We show that our method guarantees k-anonymity and that it is more efficient in terms of privacy protection and information loss than the widely used Gaussian noise addition (GNA) PPCF method. We selected GNA because it has proven to better preserve privacy by adding much less quantity of noise in large multidimensional datasets than other methods such as uniform noise addition [20]. Moreover, the experiments are performed with two well-known datasets which offer different characteristics in both data sparsity and number of dimensions. From the perspective of the SDC community, it is important to emphasise that we show that microaggregation could be used efficiently in datasets that are very sparse and also in those that are denser. More importantly, in both situations, microaggregation performs better than GNA. Finally, our proposal remains simple, facilitating its adoption.

The rest of this article is organised as follows. § 2 introduces the basic concepts on PPCF and classifies the most relevant methods. Also, some background on Statistical Disclosure Control (SDC) and microaggregation is given. Next, in § 3, we describe our PPCF method based on microaggregation. In § 4, we provide an extensive study of the results obtained by our proposal applied to well-known benchmarks. In § 5 we discuss its benefits in comparison with the Gaussian noise addition method. Finally, § 6 concludes the article and provides directions for future research.

2. Background

One of the most important limitations of CF methods is the lack of privacy. CF systems provide users with a great potential to share all types of information [13] about places to go, things to do, or products to buy, but their privacy risks are severe. The challenge is how can users contribute their personal information for CF purposes without compromising their privacy. To overcome this limitation, recent work in PPCF enables CF without leaking private information. Thus, in this section, we firstly discuss the main concepts and methods in PPCF, and secondly, we briefly introduce the principles of SDC and microaggregation, that are used in our PPCF approach.

2.1. Privacy Preserving Collaborative Filtering

The widespread use of CF on the Internet entails great opportunities for both companies and users in multiple contexts [13]. However, the lack of privacy for the contributing users is a major drawback. The relevance of privacy in CF systems is emphasised by the growing pace at which information on each user is collected and stored. Careless management of personal information, apart from being illegal in many countries, has potentially serious consequences for both the users and businesses whose information is disclosed. One of the main problems in CF is that, if customers believe their preferences/profiles may be exposed, they might decide either not to give their assessment on a particular item or to give it incorrectly or inaccurately [21]. Therefore, the feeling of poor privacy protection results in a reduction of the number and quality of evaluations.

Another drawback is that companies can acquire data about the preferences of many users in a given market, getting a big advantage over new competitors if they decide to expand into other markets. Therefore, user profiling through CF promotes in some sense monopolies. Another privacy-related drawback for users stems from the existence of large Internet quasi-monopolies, which massively gather users' preferences and may transfer them within their web of partnered companies in hardly traceable ways, leading to further user profiling.

Whilst privacy preserving CF methods obfuscate and/or hide information on user profiles, sometimes users wish to find other users having similar profiles and form a community. Indeed, communities are very usual in the Internet, but they can be a double-edged sword. On the one hand, users can conveniently obtain reliable recommendations on items from communities in a particular context. On the other hand, communities can generate a *homophily* problem in the network. More precisely, a problem of *value homophily* [22], so that recommendations outside the context of the community would give results with little sense, precisely because of the homogeneity of the group.

Privacy Preserving Collaborative Filtering (PPCF) methods aim at solving the privacy issues raised by the systematic collection of private information on preferences. Yet, privacy preservation should not prevent companies from co-operating to generate better recommendations for their customers. Due to privacy and business concerns, unprotected user data should not be disclosed between companies. In this context, data might be partitioned between various corporate 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 a $m \times n$ user-item matrix, one party A holds a subset of users $m_a \leq m$ whilst another party B holds the rest $m_b = m - m_a$; the same applies for items.

Depending on how information is stored and how recommendations are computed, PPCF schemes can be classified as *centralised* or *decentralised*. A PPCF method is called centralised if it relies on a third party to perform intermediate calculations between users or entities, or if ratings are stored in a single server where recommendations and predictions are computed. Situations in which data are partitioned as discussed above are not considered as centralised because the data are distributed between different parties. Typically, centralised PPCF methods offer higher efficiency than their decentralised counterparts since similarity and prediction computations avoid the communication overhead. However, in centralised methods, data are managed by only one party which has total control over them, with the ensuing privacy issues, if the data are not well protected at the central party. Most centralised PPCF methods add noise in several ways to perturb the data. In [23, 24] the authors propose to perturb the data by following an item-invariant Gaussian-uniform noise distribution. In the item-based PPCF scheme proposed by Zhang *et al.* in [25], perturbations are added depending on the importance of the ratings in the recommendation process. Another way to perturb data is shown in [26], where the authors propose item permutations and geometric transformations to obfuscate the data.

Regarding decentralised PPCF methods, they use members of distributed networks, considered in most cases as users, to perform intermediate calculations and predictions. Decentralised schemes generally involve less information disclosure than their centralised counterparts, but they entail the use of expensive protocols and more complex calculations. Typically, in decentralised PPCF methods, users store their own ratings. This results in a series of shortcomings like the need for the active participation of users, who are required to share their data and perform intermediate calculations. If users are not active, the amount of data over which CF is performed decreases, which in turn drastically reduces the accuracy of the recommendations.

Well-known PPCF with partitioned data schemes, which involve different parties sharing their data to perform CF with more referrals, are also regarded as decentralised methods. Several approaches with partitioned market basket databases [27] have been proposed in the literature. This kind of database is suitable to make top-N recommendations with high accuracy and low computational cost due to its binary ratings contents. In the numerical rating context, we have several approaches with partitioned data schemes such as the one proposed in [28]. Finally, schemes in which users store their own ratings can be found in [29] and [30]. For more on PPCF, the interested reader may refer to [14].

2.2. Statistical Disclosure Control and Microaggregation

Statistical Disclosure Control (SDC, [31]), also known as statistical disclosure limitation, seeks to transform microdata sets (*i.e.* datasets consisting of records corresponding to individual respondents). Such transformation is performed before publication in such a way that it is not possible to re-identify the respondent corresponding to any particular record in the anonymised published microdata set —identity disclosure— nor is it possible to discover the value of a confidential attribute (*e.g.* salary) for a *specific* respondent —attribute disclosure—.

Prior to any anonymisation process, direct identifiers (name, passport no., etc.) need of course to be suppressed from the dataset. However, some of the attributes that remain in the anonymised dataset may be *quasi-identifiers*, that is, attributes which may facilitate indirect re-identification of respondents through external data sources (available as intruders' background knowledge) that combine those attributes with direct identifiers.

In fact, in our application to PPCF, attributes will be the preferences of users on different items, and each record will collect the preferences of a particular user. We will consider all attributes to be *quasi-identifiers*, because a large number of preferences has been shown to lead to user re-identification (*e.g.* [32] identified Netflix users based on their preferences).

Microaggregation is a family of SDC algorithms for datasets, used to prevent against re-identification, which works in two stages:

1. The set of records in a dataset is clustered in such a way that: i) each cluster contains at least k records; ii) records within a cluster are as similar as possible.
2. Records within each cluster are replaced by a representative of the cluster, typically the centroid record (*i.e.* the average of the cluster).

When microaggregation is applied to the projection of records on their quasi-identifier attributes, the resulting dataset is k-anonymous, that is, to an intruder each record in the dataset is indistinguishable within a cluster of k records in terms of the quasi-identifiers. The k-anonymity property is widely accepted as a useful measure to protect privacy and we will adopt it in our proposal. However, there are other properties such as t-closeness, p-sensitivity or l-diversity that could fit in other privacy models and might deserve the attention of the PPCF community in the future.

In [33] a simple microaggregation heuristic called Maximum Distance to Average Vector (MDAV) is described, in which all clusters have exactly k records, except the last one, which might have between k and $2k-1$ records. The fact that k-anonymity only protects against identity disclosure (but not against attribute disclosure) is not a problem in our PPCF application, because all attributes are regarded as quasi-identifiers. In other words, all attributes will be modified by microaggregation to reach k-anonymity; hence, protection against attribute disclosure is also offered (in case preferences on some items are considered confidential). We will use MDAV because of its simplicity, although it has the limitation of using clusters of fixed size k.

3. Proposed Method

In this section, we describe our proposal in detail. Our scheme can be classified as a centralised PPCF method. Its architecture is shown in Figure 1.

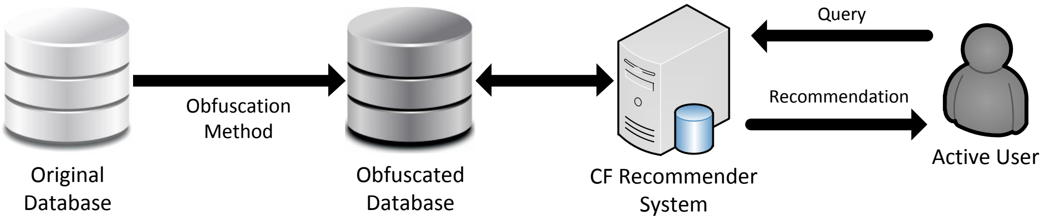


Figure 1: Centralised PPCF scheme. The user makes a request on an item to the server, which responds with a personalised prediction.

Our approach is based on the aforementioned MDAV microaggregation algorithm. However, we slightly modify MDAV in the way the leftover records are managed. The original MDAV algorithm specifies that, if at the end of the clustering process there are p records between k and $2k-1$ ($k \leq p < 2k$) that do not belong to any cluster, they should form a final cluster C_f themselves. In our approach, in order to manage the unassigned records more accurately, we first compute the mean of C_f, denoted by M_f, and we compute the distance between every C_f record and M_f. Afterwards, we compare the distance between each member of C_f and all the already formed clusters. If more than half of the records in C_f are closer to M_f than to any other cluster, we form a final cluster with the C_f elements. Otherwise, each record is added to the closest cluster amongst those already formed. Since the MDAV algorithm is a data-oriented algorithm, it cannot be proven that our approach is always better, however, in our tests, our modification has always performed better, and the improvement is specially noticeable with big values of k.

Our scheme, illustrated in Figure 2, works as follows:

1. Ensure that the dataset contains no missing values for any attribute in any record. This is necessary in order to compute the Euclidean distance between records. Imputation methods or non-personalised values can be utilised to fill the empty fields of the dataset matrix. For our experiments we have used central value imputation.

2. Once the matrix is completely filled, we compute the z-scores of each column (item) of the dataset in order to standardise the data (*i.e.* give the same statistical weight to each dimension), using the following expression
$$\text{z-score} = \frac{x_i - \mu}{\sigma}$$
where x_i is the i-th value of item x and μ and σ are the mean and the standard deviation of item x, respectively. In this way, the mean and the standard deviation of the transformed item are 0 and 1, respectively.
3. Once the standardised matrix has been obtained, we are able to apply the MDAV clustering. Users will be grouped into a number of clusters, with each cluster C_i consisting of the k most similar users, according to the Euclidean distance, where k denotes the cluster cardinality. By selecting the most similar users, we maximise the cluster homogeneity and we therefore reduce the information loss. Once the cluster relationships are established, the mean values of each C_i, denoted as M_i, are computed. Afterwards, each value of C_i is replaced by the corresponding M_i.
4. The MDAV clustering process will result in a new dataset in which members of the same cluster C_i will have the same profiles and become indistinguishable within their group. Therefore, after applying MDAV, this dataset will satisfy k-anonymity.
5. Finally, in order to make predictions, the results are de-standardised to obtain the final obfuscated dataset.

4. Experimental results

In this section, we report the experimental results of our method and compare them against those obtained with the widely used Gaussian noise addition method (GNA), which uses a Gaussian distribution with zero mean and standard deviation σ (*i.e.* $\mathcal{N}(0, \sigma)$) to perturb the dataset. Firstly, Section 4.1 shows the results related to the privacy and the utility provided by the analysed methods. Then, Section 4.2 assesses the quality of the predictions.

Experiments with GNA were repeated 50 times with each evaluated σ. As we already did in our proposal (*i.e.* Figure 2), the dataset values are standardised before the Gaussian noise is added. In order to test the quality of our method, we used two well-known CF datasets: Movielens and Jester.

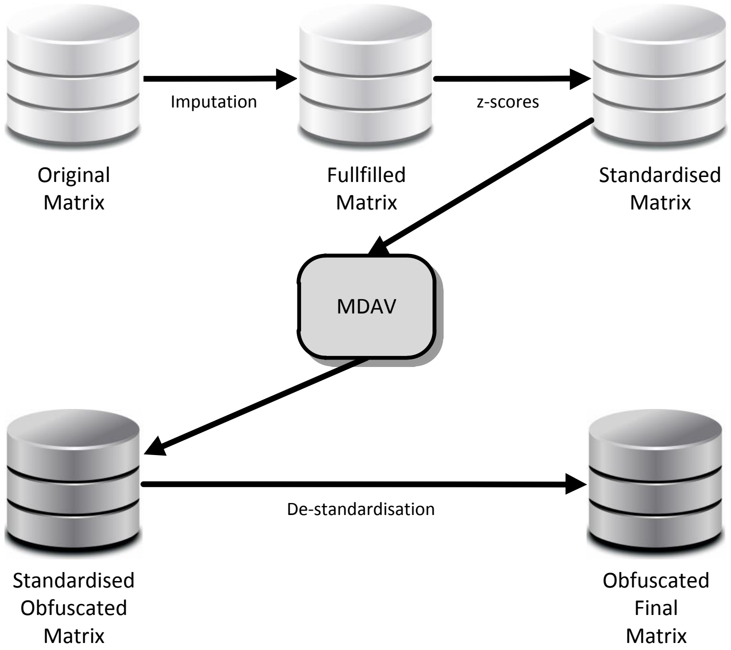


Figure 2: Graphical scheme of the proposed method.

Movielens was developed by Grouplens [4] and it is one of the most widely used reference sets in CF. Here, we consider the dataset $Movielens\ 100k$, which contains 100,000 ratings of 943 users on 1,682 films. The $Movielens100k$ range values are comprised between 1 and 5. This database is highly sparse, since more than 90% of the fields are empty. Once completely filled, the matrix contains a total of 1,586,126 values.

Jester [34] is a joke recommendation system developed in the University of California, Berkeley. The entire database has 100 jokes and ratings of 73,421 users. As a result, the matrix contains a total of 7,342,100 values. However, this database is not as sparse as $Movielens\ 100k$ since Jester has approximately 44% of empty cells.

4.1. Protection assessment: information loss and privacy

In order to measure the quality of the protection provided by a perturbation method we consider two factors, namely the information loss and the disclosure risk. The information loss is generally associated to the sum of squared errors (SSE). The SSE is commonly used as a measure of the distortion introduced on the original data. In the special case of microaggregation, the SSE is computed in vector notation as follows:

$$SSE = \sum_{i=1}^{g}\sum_{j=1}^{n_i}(x_{ij} - \bar{\mathbf{x}}_i)'(x_{ij} - \bar{\mathbf{x}}_i) \tag{1}$$

where g is the number of subsets/clusters generated by the algorithm, n_i the number of elements in each cluster, x_{ij} the vector of the j-th user of the i-th cluster and $\bar{\mathbf{x}}_i$ is the average vector of the i-th cluster. More generally, given an original dataset O represented by a matrix of $n \times m$ elements o_{ij} and a distorted/protected dataset P represented by a matrix of $n \times m$ elements p_{ij}, the SSE is computed as follows:

$$SSE = \sum_{i=1}^{n}\sum_{j=1}^{m}(o_{ij} - p_{ij})^2 \tag{2}$$

The disclosure risk (DR) measures the probability of correctly relating a record of the obfuscated/protected data matrix with a record of the original matrix. It is also known as the probability of re-identification, or the re-identification risk. For an attacker, the re-identification procedure consists in computing the distances (*e.g.* the Euclidean distance) between a given protected record p_i (corresponding to user i), and the target records o_j, that could be obtained from third party sources such as censuses and the like. In our case we assume the best scenario for an attacker (the oracle scenario) in which he has the original dataset O and the distorted dataset P and he tries to link each record p_i in P with the records o_j in O.

For each record $p_i \in P$ the attacker determines the closest record $o_j \in O$. If that closest record o_j is actually the original record belonging to p_i, the attacker succeeds and we say that p_i has been re-identified. To compute the disclosure risk, we try to re-identify all records and then compute the percentage of correct re-identifications. In terms of privacy and utility of the data, both the SSE and the DR should be low.

In the following tables and figures we show both SSE and DR results for the analysed methods: our microaggregation-based method and the GNA. Table 1 shows the results obtained with our proposed method for different values of k, which represent the cardinality of the clusters, whilst Table 2 shows the results obtained using GNA with different values of σ. It can be clearly seen that the relation between SSE and DR is much better in the

Table 1: Results of MDAV based PPCF. For the sake of clarity, the SSE results of the *Movielens* 100*k* and the Jester databases are displayed in a 10^3 and in a 10^6 scale, respectively.

	MDAV															
	k	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**25**	**50**	**75**	**100**	**150**	**200**
ML 100k	SSE	64	87	99	105	110	114	117	119	120	130	134	136	136	138	139
	DR %	40.82	26.51	19.93	15.90	12.19	12.19	9.65	7.95	7.21	2.33	0.63	0.21	0.21	0.10	0.10
JESTER	SSE	25	37	44	48	52	54	57	58	60	69	73	75	76	77	78
	DR %	47.455	30.577	22.173	17.121	13.729	11.331	9.450	8.000	6.800	1.570	0.513	0.303	0.242	0.147	0.128

Table 2: Results of GNA based PPCF. For the sake of clarity, the SSE results of the *Movielens* 100*k* and the Jester databases are displayed in a 10^3 and in a 10^6 scale, respectively.

	GNA																	
	σ	**0.25**	**0.5**	**0.75**	**1**	**1.25**	**1.5**	**1.75**	**2**	**2.5**	**3**	**3.5**	**4**	**5**	**10**	**20**	**40**	**50**
ML 100k	SSE	246	248	257	275	302	336	376	418	509	592	663	727	830	1078	1221	1294	1339
	DR%	100	100	98.51	89.28	68.50	50.58	44.53	27.99	18.76	10.49	8.58	7.21	4.24	1.40	0.42	0.31	0.10
JESTER	SSE	6	25	55	93	136	181	226	268	342	404	454	495	558	698	774	813	821
	DR%	99.873	95.298	78.270	58.528	42.056	30.221	22.030	15.759	8.127	4.193	2.255	1.301	0.463	0.036	0.006	0.001	0

MDAV-based approach for the analysed databases. Actually, note that since our method satisfies k-anonymity, by design its DR is upper-bounded by $1/k$.

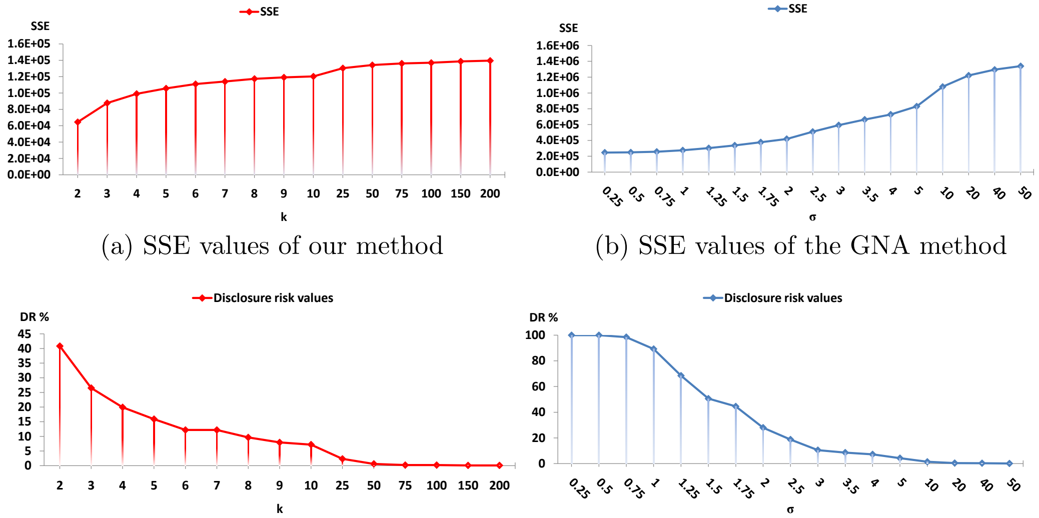
4.1.1. Movielens 100k

Figure 3a and Figure 3c respectively show the SSE and DR for the MDAV-based PPCF for different values of k. It can be observed that their behaviour is pretty antagonistic. When SSE is increased, DR is reduced accordingly.

Figure 3b and Figure 3d show SSE and DR for the GNA approach respectively. Similarly to the MDAV-based approach, when SSE grows, DR decreases. However, as we discuss in Section 5, the GNA method needs to add much more distortion to the data (*i.e.* SSE is increased) than the MDAV-based approach to reach the same DR.

4.1.2. Jester

The behaviour of the results obtained for the Jester database, illustrated in Figure 4, is almost identical to the one for the *Movielens* 100*k* database. Likewise, data remain affected in the same way, but in another scale, due to the range of values of this database. Since the σ values proposed for the GNA approach are the same in both matrices, the amount of added noise has a lower impact in Jester database because the range of possible values is significantly wider. In the most extreme case (*i.e.* σ=50), it may be observed



(a) SSE values of our method

(b) SSE values of the GNA method

(c) DR values of our method

(d) DR values of the GNA method

Figure 3: SSE and DR results of the implemented methods on the *Movielens* $100k$ database.

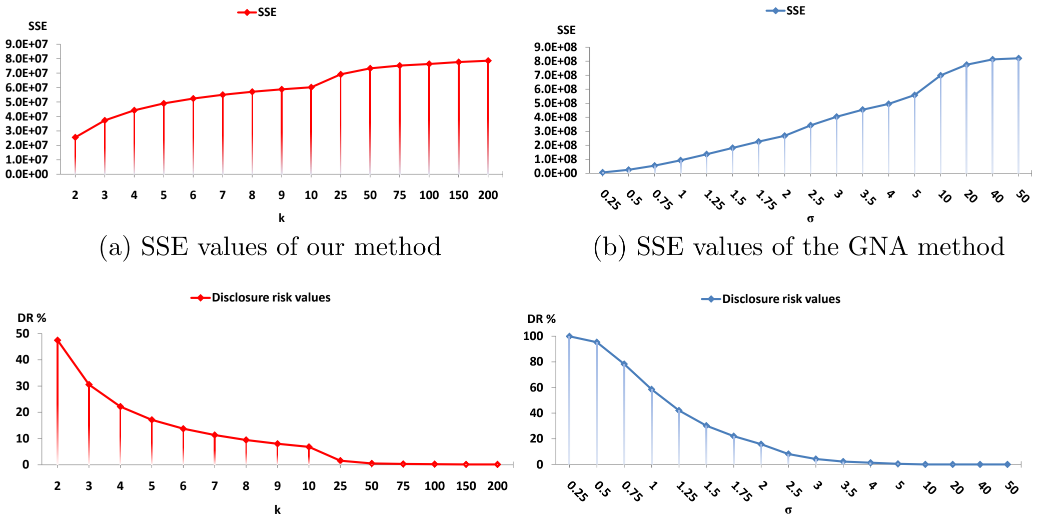
that the obtained DR is equal to 0. However, the SSE is so high (8.21×10^8) that data are practically useless.

4.2. Prediction accuracy

In the previous section, we have analysed the information loss SSE and the disclosure risk DR provided by the MDAV-based and the GNA approaches. However, information loss is not entirely captured by SSE. Note that the protected data will be used by recommender systems to make predictions about which items a user would be more interested in. Thus, it is important to check how accurate are predictions after protecting the data.

In order to measure the aforementioned prediction accuracy, we have defined a training set with 80% of the item values and a test set with the remaining 20%, for both databases. We will use the protected records for the users in the training set and the original records for the users in the test set. The predictions are computed as follows:

- *Find closest neighbour.* Given a user U_i in the test set, find its closest user, say U_j, in the protected training dataset.



(a) SSE values of our method

(b) SSE values of the GNA method

(c) DR values of our method

(d) DR values of the GNA method

Figure 4: SSE and DR results of the implemented methods on the Jester database.

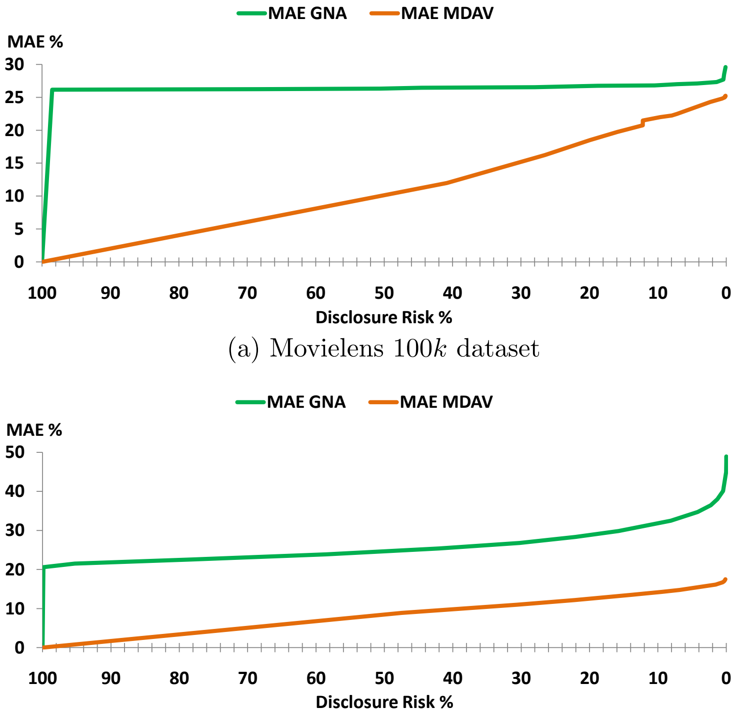
- *Assign prediction.* The predicted values for user U_i are those that correspond to U_j in the test set.

Once the prediction for all users in the test set has been done as above, we compute the error between the original values of the test set and the values assigned by the above procedure. To compute this error we apply the widely used mean absolute error (MAE), defined as follows:

$$MAE = \frac{\sum_{i=1}^{n} |p_i - r_i|}{n} \tag{3}$$

where n is the number of predicted elements, p_i is the predicted value for element i, and r_i is the real value of i in the test set. The MAE outcomes in respect of the DR are displayed in Figure 5.

As depicted in Figure 5a, initially the MAE outcomes of the GNA method grow significantly with low values of added noise as a result of the small range of the Movielens 100k's values. Moreover, such range of values truncate the noise rapidly once a considerable amount of noise is reached. Additionally, the sparseness of the matrix negatively affects the resulting MAE. Finally, the growth increases as the value of σ does. The growth of the MAE in our



(a) Movielens 100k dataset

(b) Jester dataset

Figure 5: Relation between MAE and DR for the analysed methods on the selected databases. (The lower the better)

method is linear with respect to k and achieves significantly lower values, which means recommendations of better quality.

Regarding the Jester database (Figure 5b), the MAE achieved with our method is better than the one achieved with Movielens 100k because its lower sparseness. However, the range of the Jester dataset is wider and then the noise introduced by GNA affects data with a slower pace.

Moreover, to perform a clearer comparison between our proposal and GNA, we have selected obfuscated datasets with the same privacy level (*i.e.* DR). Thereafter, for the sake of simplicity, we have performed a single comparison for both databases. Note that any DR value could have been chosen as long as it is the same for both methods. The results of such comparison are displayed in Table 3.

Table 3: Example of a comparison of MAE and %MAE results between GNA and MDAV for the two analysed datasets. The lower the better.

Database	**Method**	**MAE**	**%MAE**
ML 100k	$MDAV,\ k = 10$	0.89	22.25
	$GNA,\ \sigma = 4$	1.08	27
Jester	$MDAV,\ k = 9$	2.91	14.55
	$GNA,\ \sigma = 2.5$	6.50	32.5

5. Comparison and Discussion

In the previous section we presented the results obtained by our microaggregation-based approach and the classical GNA approach. In this section we briefly compare these results and show that the MDAV-approach is superior, both in terms of privacy and prediction accuracy.

5.1. Movielens 100k

In Figure 6a, we can see a comparison between SSE and DR for both methods. In the X-axis we represent DR and in the Y-axis we show SSE. This figure can be used to read the amount of noise, in terms of SSE, that is required by each method to achieve a given DR. For example, it can be seen that for a fixed value of $DR = 30\%$ the MDAV-approach roughly introduces an error in the order of $100K$ whilst the GNA approach requires $400K$.

The smallest possible DR value, if the data is obfuscated with the MDAV method, is $\frac{1}{943} \simeq 0.1\%$ for the $Movielens$ $100k$ dataset. In order to obtain such value, the MDAV-approach needs to form clusters of $k = 150$ elements, which leads to an SSE of $138,650$. In contrast, the GNA obtains such DR value with an SSE of $1,339,008$, which is almost one order of magnitude larger.

These results clearly show that the proposed approach perturbs data in a much more efficient way. Moreover, as already mentioned, our method provides k-anonymity and therefore upper-bounds the disclosure risk by design.

Regarding the quality of predictions, Figure 5 shows the relation between MAE and DR for both datasets. It may be observed that the growth resembles that of Figure 6, with slight differences due to the % of the prediction set and the rating's truncation. Moreover, Table 3 shows an example of the predictions accuracy. It may be seen that when the predictions are conducted using the data protected with MDAV, MAE is 22.25% with a DR value of 7.21%, which is a considerable privacy level. On the contrary, the values predicted by using the data protected with GNA lead to a 27% MAE, which is almost 5 percentage points higher. Therefore, we may conclude that both the quality of the predictions and the quality of privacy are better in our method based on MDAV.

5.2. Jester

In Figure 4d and in Table 2 we notice that DR reaches a considerable privacy level with low values of σ in the case of the Jester database. Although the growth pace of the SSE is nearly the same for both databases, the quantity of noise added in the $Movielens$ $100k$ is much lower due to its lower value range, compared with the Jester database, especially for high values of σ.

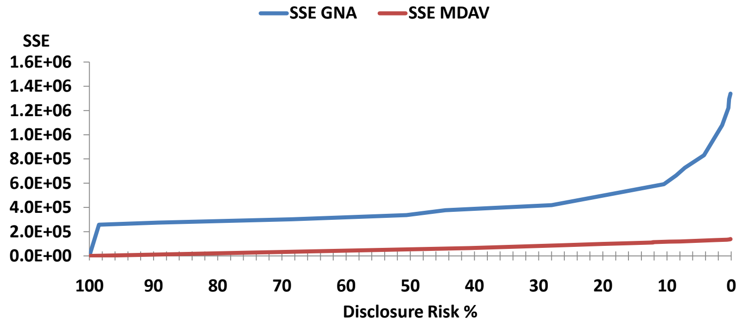
The quality of the predictions for the selected obfuscated datasets is shown in Table 3. For almost the same level of privacy, the GNA approach reaches a 32.5% MAE, which is more than twice the MAE achieved by the MDAV method (*i.e.* 14.55%). Such accuracy of the MDAV proposal is mainly due to the density of the Jester database, which contains more than 55% of original votes.

In Figure 6b we can observe the efficiency of the applied noise in the Jester database. The behaviour is nearly identical to the one shown in Figure 6a for $Movielens$ $100k$, except for the initial growth of SSE with the GNA method. This growth is produced because the initial values of σ perturb less the Jester data due to their wider range of values, compared with the $Movielens$ $100k$'s range.

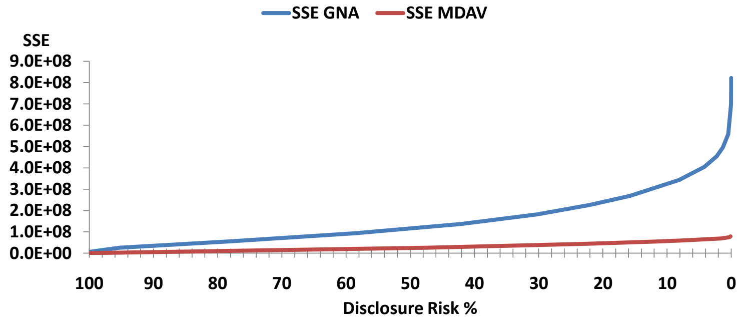
Both Figures 6a and 6b confirm that our method is applicable to sparse and dense databases and its quality is far higher than the one of GNA, regardless of the data. Obviously, the quality of the predictions is highly related with the density of the database.

6. Conclusions

Collaborative Filtering is a recommender system used to perform automatic recommendations to users in multiple contexts. Despite the great advantages of using CF, we have highlighted its downside regarding users' privacy. Although a large amount of CF methods have been proposed, more



(a) Movielens 100k dataset



(b) Jester dataset

Figure 6: Relation between SSE and DR for the analysed methods on the selected databases.

study is still needed as there are many challenges to overcome. Probably, the most significant amongst them is the proper protection of users' privacy.

Definitely, there is a trade-off between the privacy of users' preferences and the quality of the recommendations obtained. Therefore, in this article, we have proposed a novel PPCF method based on microaggregation. The results obtained over the evaluated databases demonstrate that the proposed method perturbs data in a much more efficient way than other well-known methods such as GNA. Moreover, our proposal achieves *k-anonymity*, which guarantees privacy by design, a feature not offered by GNA.

It is important to emphasise that our proposal concentrates on the protection of the data and it might be combined with other techniques such as the encryption of identifiers to provide a holistic protection of the users' privacy. Future work will focus on two different directions. The first one is to improve the efficiency of our method in order to enable implementation in a decentralised setting. Certainly, a decentralised version of MDAV could be achieved by using homomorphic encryption or other well-known privacy preserving data mining techniques. However, boosting its performance to make it a real choice in practice is an open issue that we plan to study in the near future. The second direction regards the centralised setting, where more computational complexity can be tolerated; hence, in this case, MDAV could be replaced by microaggregation heuristics with variable group size [35], [36] in order to reduce information loss as much as possible.

Acknowledgments and disclaimer

This work was partly supported by the Government of Catalonia under grant no. 2014 SGR 537. Dr. Domingo-Ferrer is partially supported as an ICREA Acadèmia researcher by the Government of Catalonia, and by the EC under FP7 projects"DwB" and "Inter-Trust". Dr. Solanas is partly funded by La Caixa Foundation through project "SIMPATIC" RECERCAIXA'12. The authors with the UNESCO Chair in Data Privacy are solely responsible for the views expressed in this article, which do not necessarily reflect the position of UNESCO nor commit that organisation.

References

[1] P. Resnick, H. Varian, Recommender systems, Communications of the ACM 40 (3) (1997) 56–58.

[2] H. Jafarkarimi, A. Sim, R. Saadatdoost, A Naive Recommendation Model for Large Databases, International Journal of Information and Education Technology 2 (3) (2012) 216–219.

[3] D. Goldberg, D. Nichols, B. M. Oki, D. Terry, Using collaborative filtering to weave an information tapestry, Communications of the ACM 35 (12) (1992) 61–70.

[4] P. Resnick, N. Iacovou, M. Suchak, GroupLens: an open architecture for collaborative filtering of netnews, Proceedings of the ACM CSCW (3) (1994) 175–186.

[5] J. L. Herlocker, J. a. Konstan, L. G. Terveen, J. T. Riedl, Evaluating collaborative filtering recommender systems, ACM Transactions on Information Systems 22 (1) (2004) 5–53.

[6] X. Su, T. M. Khoshgoftaar, A Survey of Collaborative Filtering Techniques, Advances in Artificial Intelligence 2009 (3) (2009) 1–19.

[7] F. Cacheda, V. Carneiro, D. Fernández, V. Formoso, Comparison of collaborative filtering algorithms, ACM Transactions on the Web 5 (1) (2011) 1–33.

[8] B. Sarwar, G. Karypis, J. Konstan, J. Reidl, Item-based collaborative filtering recommendation algorithms, Proceedings of the 10th international conference on World Wide Web (15) 285–295.

[9] D. Lemire, A. Maclachlan, Slope one predictors for online rating-based collaborative filtering, Society for Industrial Mathematics 05 (12) (2005) 471–475.

[10] A. Bilge, H. Polat, A comparison of clustering-based privacy-preserving collaborative filtering schemes, Applied Soft Computing 13 (5) (2013) 2478–2489.

[11] D. Y. Pavlov, D. M. Pennock, A Maximum Entropy Approach to Collaborative Filtering in Dynamic, Sparse, High-Dimensional Domains, Advances in Neural Information Processing Systems 15 15 (2/3) (2003) 1441–1448.

[12] A. Schwaighofer, V. Tresp, H. Kriegel, Probabilistic memory-based collaborative filtering, IEEE Transactions on Knowledge and Data Engineering 16 (1) (2004) 56–69.

[13] Y. Shi, M. Larson, A. Hanjalic, Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges, ACM Computing Surveys (CSUR) 47 (1) (2014) 1–45.

[14] F. Casino, C. Patsakis, D. Puig, A. Solanas, On privacy preserving collaborative filtering: Current trends, open problems, and new issues, in: IEEE 10th International Conference on e-Business Engineering (ICEBE), 2013, pp. 244–249.

[15] W. Wu, J. Zhou, Y. Xiang, L. Xu, How to achieve non-repudiation of origin with privacy protection in cloud computing, Journal of Computer and System Sciences 79 (8) (2013) 1200–1213.

[16] A. Wahid, C. Leckie, C. Zhou, Estimating the number of hosts corresponding to an intrusion alert while preserving privacy, Journal of Computer and System Sciences 80 (3) (2014) 502–519.

[17] X. Zhang, C. Liu, S. Nepal, J. Chen, An efficient quasi-identifier index based approach for privacy preservation over incremental data sets on cloud, Journal of Computer and System Sciences 79 (5) (2013) 542–555.

[18] A. Martinez-Balleste, P. A. Pérez-Martínez, A. Solanas, The pursuit of citizens' privacy: a privacy-aware smart city is possible, Communications Magazine, IEEE 51 (6).

[19] F. Casino, J. Domingo-Ferrer, C. Patsakis, D. Puig, A. Solanas, Privacy preserving collaborative filtering with k-anonymity through microaggregation, in: IEEE 10th International Conference on e-Business Engineering (ICEBE), 2013, pp. 490–497.

[20] C. C. Aggarwal, On Randomization, Public Information and the Curse of Dimensionality, 2007 IEEE 23rd International Conference on Data Engineering (2007) 136–145.

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

[22] M. McPherson, L. Smith-Lovin, J. M. Cook, Birds of a Feather: Homophily in Social Networks, Annual Review of Sociology 27 (1) (2001) 415–444.

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

[24] H. Polat, L. Hall, SVD-based Collaborative Filtering with Privacy, Proceedings of the 2005 ACM symposium on Applied computing (2005) 791–795.

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

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

[27] I. Yakut, H. Polat, Estimating NBC-based recommendations on arbitrarily partitioned data with privacy, Knowledge-Based Systems 36 (2012) 353–362.

[28] I. Yakut, H. Polat, Arbitrarily distributed data-based recommendations with privacy, Data & Knowledge Engineering 72 (2012) 239–256.

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

[30] C. Kaleli, H. Polat, P2P collaborative filtering with privacy, Turkish Journal of Electric Electrical Engineering and Compute Science 18 (1) (2010) 101–116.

[31] A. Hundepool, J. Domingo-Ferrer, L. Franconi, S. Giessing, E. Schulte-Nordholt, K. Spicer, P.-P. de Wolf, Statistical Disclosure Control, Wiley, 2012.

[32] A. Narayanan, V. Shmatikov, Robust de-anonymization of large sparse datasets, in: 29th IEEE Symposium on Security and Privacy, 2008, pp. 111–125.

[33] J. Domingo-Ferrer, V. Torra, Ordinal, continuous and heterogenerous k-anonymity through microaggregation, Data Mining and Knowledge Discovery 11 (2) (2005) 195–212.

[34] D. Gupta, M. Digiovanni, H. Narita, K. Goldberg, Jester 2.0: Evaluation of a new linear time collaborative filtering algorithm, 22nd International ACM SIGIR (1999) 291–292.

[35] A. Solanas, U. Gonzalez-Nicolas, A. Martinez-Balleste, A variable-mdav-based partitioning strategy to continuous multivariate microaggregation with genetic algorithms, in: Neural Networks (IJCNN), The 2010 International Joint Conference on, IEEE, 2010, pp. 1–7.

[36] A. Solanas, A. Martinez-Balleste, V-MDAV: a multivariate microaggregation with variable group size, in: 17th COMPSTAT Symposium of the IASC, Rome, 2006, pp. 917–925.