Journal of Universal Computer Science, vol. 23, no. 2 (2017), 146-166
submitted: 30/6/16, accepted: 20/1/17, appeared: 28/2/17 © J.UCS

A Method for Privacy-preserving Collaborative Filtering Recommendations

Christos K. Georgiadis
(University of Macedonia, Thessaloniki, Greece
geor@uom.edu.gr)

Nikolaos Polatidis
(University of Macedonia, Thessaloniki, Greece
npolatidis@uom.edu.gr)

Haralambos Mouratidis
(University of Brighton, Brighton, United Kingdom
H.mouratidis@brighton.ac.uk)

Elias Pimenidis
(University of the West of England, Bristol, United Kingdom
Elias.pimenidis@uwe.ac.uk)

Abstract: With the continuous growth of the Internet and the progress of electronic commerce the issues of product recommendation and privacy protection are becoming increasingly important. Recommender Systems aim to solve the information overload problem by providing accurate recommendations of items to users. Collaborative filtering is considered the most widely used recommendation method for providing recommendations of items or users to other users in online environments. Additionally, collaborative filtering methods can be used with a trust network, thus delivering to the user recommendations from both a database of ratings and from users who the person who made the request knows and trusts. On the other hand, the users are having privacy concerns and are not willing to submit the required information (e.g., ratings for products), thus making the recommender system unusable. In this paper, we propose (a) an approach to product recommendation that is based on collaborative filtering and uses a combination of a ratings network with a trust network of the user to provide recommendations and (b) "neighbourhood privacy" that employs a modified privacy-aware role-based access control model that can be applied to databases that utilize recommender systems. Our proposed approach (1) protects user privacy with a small decrease in the accuracy of the recommendations and (2) uses information from the trust network to increase the accuracy of the recommendations, while, (3) providing privacy-preserving recommendations, as accurate as the recommendations provided without the privacy-preserving approach or the method that increased the accuracy applied.

Keywords: Collaborative Filtering, Trust Network, Privacy, Recommender Systems
Categories: H.3.3, H.3.5, K.4.1

1 Introduction

Recommender Systems are information systems algorithms that are used to cope with the information overload problem on the Internet [Jannach, Zanker, Felfernig and Friedrich 2010, Polatidis and Georgiadis 2013, Valcarce, Parapar and Barreiro 2015]. Their most common use can be found in e-commerce sites such as Amazon.com and Epinions, where the user is overwhelmed with too much information and it is very difficult to make a choice about an item or a topic that suits her respective needs. Recommender systems are intelligent systems that have been proposed and used by e-commerce vendors in a variety of environments, including mobile platforms. Their job is to propose to the user the most relevant items or services, according to the current conditions and context. The recommendations are retrieved according to a specified set of rules usually set by the system itself according to a specific user's behaviour and characteristics.

Collaborative filtering is the most widely used type of algorithm for providing personalized recommendations in e-commerce environments [Polatidis, Georgiadis, Pimenidis and Mouratidis 2017, Shi, Larson and Hanjalic 2014]. The algorithm makes suggestions of items similar to the users' preferences as found in their rating history. Knowledge based filtering algorithms base their operation on user provided data, such as preferences and choices and by asking the user to provide specific information. Then the algorithm provides the recommendations according to a specified set of rules. Social media recommendation systems is an active field of study and standards have not been defined yet [Tang, Hu and Liu 2013]. However, other researchers such as [Carmagnola, Vernero and Grillo 2014] state that, social media recommender systems can give acceptable results when compared to pure collaborative filtering that are based on ratings only. Controversies exist mainly about a) what kind of data this kind of algorithms will take into consideration and include, b) what the user likes or dislikes, and c) what various teams of friends think that a user will like or dislike. Hybrid recommender systems are combinations of two or more traditional recommendation algorithms in order to provide more accurate results [Burke 2002]. Hybridization can be achieved in different ways such as combining the results of each algorithm in one interface or using the output of one as the input of another.

Earlier studies have shown that the problem of making the right recommendation is very difficult to solve with a single algorithm [Bogers and van den Bosch 2011]. We also believe that if we provide a number of different recommendation approaches and provide users (administrators and/or end-users) with the option to switch between them, we can produce better and more accurate results. [Burke 2002] presented different methods whereby algorithms can be combined to create hybrid systems and get improved results. Two of the most significant methods are namely the mixed fusion method (which combines all the outputs of different algorithms into a single top N recommendation list) and the switching algorithm (which can change between different individual algorithms according to certain criteria). An equally significant method is the meta-level hybrid method where the output of one algorithm is provided as an input to another. The only drawback of the latter is that it does not take into consideration data from social networks. In fact, social rating networking sites such as Epinions have started to attract many people. In such networks, people can register

and add other people as friends and also rate products [Massa and Avesani 2007]. In this paper, we utilize the data from such networks to validate our proposed method.

Privacy concerns play a crucial role when users come to the point where they must disclose information. Privacy is the right to keep your life private, therefore data privacy is the power, control or ability that a user has about the way her data will be processed. Furthermore, data privacy is tightly related to technologies that can be applied to keep personal user information private. In recommender systems, users that are privacy-aware are divided into three categories [Jeckmans et al. 2013]:

- Users that will accept to supply any kind of personal information to a vendor in exchange for personalized content.
- Users that will supply a certain amount of information to a vendor to receive improved personalized recommendations.
- Users that do not accept to give any kind of information due to privacy concerns.

According to [Chellappa and Sin 2005] the user perceived value for personalization is very high. It is a very important factor that could change the mind of a user about privacy concerns. At the same time, a user wants to be in control of how her data will be used. Other factors include trust towards a vendor, positive past experiences and the overall vendor reputation.

1.1 Contributions

Recommender systems have advanced through academic research and commercial development to a point where their scope is well-known and understandable, but certain limitations, such as privacy concerns, have restricted their use. The main research question of our work is how to exploit significant information from social networks (user ratings for products and services as well as trust information about users) to deliver as accurate as possible recommendations to the user, without compromising their privacy. Thus, we propose a privacy model that can be applied to protect the data from unauthorized access. We also propose a trust based collaborative filtering method that combines similarity matrices from the user rating network and the trust network. The following contributions that advance the field have been made and are summarized below:

1. A recommendation approach that is based on Collaborative Filtering algorithm in combination with the trust network of the user is proposed.
2. A privacy-preserving approach that is applied at the server side to offer an extra level of protection to its users is employed. The proposed approach is based on a modified role-based access control model to prevent unauthorized internal access to user data. Furthermore, a randomization function is applied to user similarity values before the generation of the recommendations.

3. The proposed approaches have been experimentally evaluated and are shown to be both practical and effective.

The rest of this paper is organized as follows: Section 2 gives an overview of how the standard user-rating collaborative filtering works. Section 3 gives an overview of how trust-aware recommenders work. Section 4 gives a description of our proposed method. In section 5 we present and discuss the experimental evaluation of our approach. In section 6 the related work part is found and section 7 presents the conclusions of the paper.

2 Rating Prediction based on User-item Ratings Network

Collaborative Filtering (CF) is an approach where information from previous opinions is exploited and the ratings of an existing user base are used to predict which items the current user will probably like [Shi et al. 2014]. An example of such a user network is shown in Figure 1, where we can see users assigning values to products.

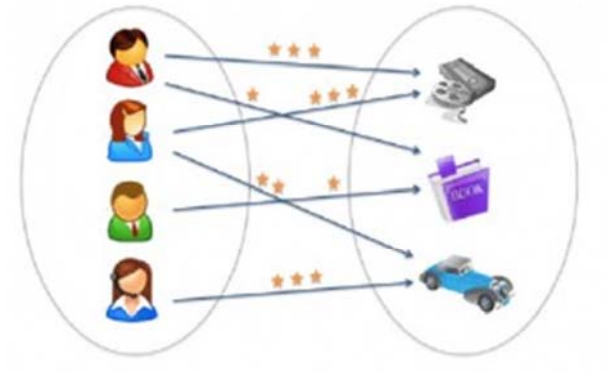

Figure 1: Typical ratings in collaborative filtering

In this approach a nearest neighbour recommendation database is created. The idea is very simple and is the following: a database of users, products and ratings is stored in the system and an algorithm is used to identify users that had similar tastes in the past to those of the current user. Then for every product that has not been rated by the user, a rating is computed based on ratings from users with similar history. Consider the following example where the database of a virtual system is represented in Table 1.

	Product1	Product2	Product3	Product4	Product5
User1	5	3	4	4	Empty
User2	3	1	2	3	3
User3	4	3	4	3	5
User4	3	3	1	5	4
User5	1	5	5	2	1

Table 1: A ratings database

The next step in the recommendation process is to use a recommendation similarity measure such as Pearson correlation or Cosine similarity. We use the Pearson correlation similarity which is defined in equation 1, where Sim (a, b) is the similarity of users a and b, $r_{a,p}$ is the rating of user a for product p and $r_{b,p}$ is the rating of user b for product p. P is the set of all products. Then, the similarity matrix SimCF, is calculated.

$$Sim\ (a,b) = \frac{\sum p\in P(ra,p-\bar{r}a)(rb,p-\bar{r}b)}{\sqrt{\sum p\in P(ra,p-\bar{r}a)^2}\sqrt{\sum p\in P(rb,p-\bar{r}b)^2}} \quad (1)$$

For example, the similarity matrix for User1 is shown in Table 2. The similarity values are in the range of -1 (worst) to 1 (best) in such systems. Therefore, the user closest to User1 is User3 and the rating prediction for Product5 is derived from User3 and the value is 5.

	User1	User2	User3	User4	User5
User1	1	0.70	0.85	0.2	-0.79

Table 2: Similarity table

3 Rating Prediction based on User Trust Network

In social rating networks users, can form friendship networks, where they can denote who they trust. In such networks nodes and edges are formed, with every edge being a pair of nodes [Symeonidis, Tiakas and Manolopoulos 2011]. Table 3 is an n x n matrix which represents who trusts whom. We assume that the network is directed and value 1 denotes trusts whereas '–' denotes no trust and no negative impact.

	User1	User2	User3
User1	-	1	-
User2	-	-	1
User3	-	1	-

Table 3: Who trusts whom representation

The next step in the process is to either take into consideration only the users' trust network and retrieve recommendations from them alone, or, as an alternative approach, to use a similarity measure (such as Jaccard Coefficient, Adamic/Adar, Katz and Common Neighbors Index) in order to calculate the similarity of values [Symeonidis et al. 2011]. By using the trust network a propagation model is applied in order to deliver a neighborhood of users with the highest degree of similarity [Goldberg, Roeder, Gupta and Perkins 2001, Massa and Avesani 2007]. Then, a similarity table like the one in Table 4 is produced and recommendations are derived based on information from the neighbors with the higher similarity value.

	User1	User2	User3
User1	-	0.65	-
User2	-	-	0.88
User3	-	0.30	-

Table 4: User network similarity

4 An Integrated Recommendation Approach

We propose the combination of a hybrid recommendation approach with the use of a privacy preservation approach to protect user privacy and maintain high accuracy.

The proposed integrated approach contains the following elements:

1. A modified role-based access control model to prevent unauthorized internal access to user data.

2. A randomization approach that modifies the similarity values between users that form the nearest neighbours. The server applies this approach to deliver relevant but different recommendations to the requesting user. By utilizing such an approach, it becomes harder for an intruder to guess the neighbours of the current user. More specifically, every time a user requests recommendations a neighbourhood of similar users is created. The randomization approach makes sure that the order of the neighbourhood is different each time; thus, making it harder to guess the neighbourhood and inject false ratings to the server.

3. The randomization of the similarity values has the effect of reducing the accuracy of the recommendations. Thus, information from the trust network is utilized to compensate the accuracy.

Collaborative Filtering with incorporated trust (CFTrust) is a hybrid method that improves the quality of recommendations by incorporating information from the users' trust network into the rating network. Monolithic hybridization [Jannach et al. 2010] is used, as there is one recommendation engine which however is based on different sources. Our approach aims to modify the base of the collaborative filtering method to get recommendations of better quality. We calculate the similarity values using standard collaborative filtering as described in section 2 to produce the similarity matrix based on the user-rating network. Subsequently we define a trust-aware similarity method that is based on [Symeonidis et al. 2011]. Our method differs since it goes down two levels and assumes that a friend of a friend can be trusted. The method is applied to the trust network, following the steps described in section 3, and produces a similarity matrix. Equation 2 shows our proposed similarity method between a1 and b1. The values of CFtrust are between -1 and 1. In the case of a value being below -1 this is set to -1 while at the higher end of the spectrum the maximum value is always set to 1.

$$\boldsymbol{SimTA}\ (\boldsymbol{a1}, \boldsymbol{b1}) = \begin{cases} \boldsymbol{0.50} & (\boldsymbol{if\ friend}) \\ \boldsymbol{0.375} & (\boldsymbol{if\ friend\ of\ a\ friend}) \\ \boldsymbol{0} & (\boldsymbol{otherwise}) \end{cases} \quad (\mathbf{2})$$

In our approach, we use 0.50 for friends and 0.375 for friends of friends as fixed similarity values. These values are then added to those obtained from the traditional collaborative filtering method to produce a final similarity table which will give a higher similarity value to friends and to a less extend to friends of friends. The recommendations provided by this approach increase the accuracy as explained in section 5. The final similarity table is derived from the user-rating network similarity table with the trust-based similarity table such asusing the following formula.

$$CFTrust = SimCF + Trust$$

Note that *SimCF* is the user-rating network similarity table, which contains the values produced by following the steps explained in section 2, while *Trust* is the trust-based similarity table, which contains the values produced by following the steps explained in section 3. The trust table can be produced by using a common similarity function as described in section 3, however in our case we use our similarity function *SimTA (a1, b1)*, where the values of 0.50 have been assigned to the friends of the user and we assigned 0.375 to the friend of a friend, because is still an important parameter but of less significance (thus the value of 0.50 of a friend was downgraded by 25%). Note that the suitability of these values is verified by the experimental evaluation for the Epinions dataset [Massa and Avesani 2007].

4.1 Optional similarity table weight

Two weight variables *w1*, and *w2* are defined. Both take values from 0 to 1. If the value is smaller, then, the weight on the similarity table is smaller and the higher the value is then the weight of the similarity table becomes higher. These two weight variables give us the flexibility to adjust the relative importance between the traditional collaborative filtering similarity (SimCF) and the proposed similarity (Trust). To give a lower or higher weight to a similarity table a human expert needs to decide how the hybrid method will work and give more weight to the similarity values provided by the traditional collaborative filtering method or to those by the trust network. This is achieved by multiplying the matrix with the weight variable, such as:

$$w1 \cdot SimCF + w2 \cdot Trust$$

The use of weight variables has been proposed before [Massa and Avesani 2007][Symeonidis et al. 2011]. However, in our approach we propose both the use of static values for the variables, such that can be adjusted by the user, and the use of randomization w1 from 0 to 1 [0...1] and w2 from 0 to 1 [0… 1] values to address the diversity of the recommended products. For example, if a larger value is generated for the rating-similarity network it means that an extra weight is put on the products recommended from that network and the same applies to the user-trust network for larger values of that similarity matrix.

If the value of *w1* in our example remains 1 this implies that the values of the similarity ratings matrix remain on the 100% of their value. As the value decreases it means that the values are decreased. For example, a value of 0.5 means 50% less in each value. Moreover, if the value of *w2* in our example remains 1 this implies that the values of the similarity trust matrix remain on the 100% of their value. As the value decreases it means that the values are decreased. For example, a value of 0.5 means 50% less in each value. If we want to give extra weight in a single similarity measure, then we decrease the value of the variable of the other similarity measure. For example, a value of 0 in the second table means that the algorithm will use only the user ratings matrix. The weights on similarity tables can be used to solve the cold start problem to some extent.

4.2 Privacy-aware role based access control for recommender systems

Privacy in recommender systems is a crucial factor that has an impact on the accuracy of the system [Toch, Wang and Cranor 2012]. Although the existing methods do their job well by protecting the user privacy to a certain level, the accuracy of the predictions is questionable. Cryptography-based existing approaches are not flexible, providing full or no privacy. Moreover, previous researchers are focused on the communication level and privacy, both at the client side. Finally, the use of trusted third party servers has been proposed in the methods mentioned above. We describe privacy protection as a method that can be fragmented into several parts, thus

satisfying every user need. Our model is based on an extended privacy aware access control and on neighborhood randomization.

We propose and define a usable privacy-aware access control model, which can be applied in the server's database to protect user privacy, while releasing any necessary data to operators that have access to a certain level in the hierarchy. A key success factor of a multi-level approach to user privacy is the number of levels and generalization methods employed, with the aim to remain adequate and usable. Many people are concerned that their data will be used in ways different than those intended. That's the main reason why most e-commerce, m-commerce and various web based systems encounter losses in both profit and volume of service provision.

4.2.1 Definitions

To support the privacy-aware model we need to take into consideration and define what data need to be accessed and for what purpose that data need to be accessed. Based on the Role Based Access Control (RBAC) model [Sandhu, Coyne, Feinstein and Youman 1996] we present a definition of our access control model, which extends the core P-RBAC of the Privacy-aware role-based access control [Ni et al. 2010].

The core P-RBAC is composed of the following elements:

- A set *U* of users, a set *R* of roles, a set *D* of data, a set *Pu* of purposes, a set *A* of actions, a set *O* of obligations, and a condition language *LC0*..
- The set of *Data Permissions* $DP = \{(a, d) \mid a \in A, d \in D\}$.
- The set of *Privacy-sensitive Data Permission* $PDP = \{(dp, pu, c, o) \mid dp \in DP, pu \in Pu, c \text{ is an expression of } LC0, o \in P(O)\}$. $P(O)$ is the power set of *O*.
- *User Assignment* $UA \subseteq U \times R$, a many-to-many mapping user to role assignment relation.
- *Privacy-sensitive Data Permission Assignment* $PDPA \subseteq R \times PDP$, a many-to-many mapping of privacy-sensitive data permission to role assignment relation.

We tailor the characteristics of core P-RBAC model as follows:

- Our proposed security policy states that an *authorization* is given to a *Software Agent(S)* or a *person (P)* only if he/it is in possession of the required hierarchy level for the *purpose* the data are requested and within the required time interval.
- Software agents, need to be controlled for their access actions, regarding sensitive data, thus we define a specific role: Agent$\in$R
- A set of adequate policies *PO = {PO1… POm},* both for software and human roles are used to represent the rules when specific access is required; formally defined as $PO \subseteq R\ X\ Pu$. Table 5 shows an example of an access control policy.

```
IF
    PO(P1) ∧ Role (r) ∧ RoleEquals (r,"agent") ∧ Pu (r,"recommendation")
THEN
    canAccess (u,D)
```

Table 5: An Example Access Policy

4.3 Matrix randomization for neighbourhood privacy

We use data randomization to generate a random *n x n* matrix rm with mrandom values between -t to t [-t…t] with Table 6 showing such an example. The rm matrix is added to the CFTrust matrix to create a new neighborhood privacy-preserving matrix, such as:

New privacy-preserving matrix = CFTrust + rm

	User1	User2	User3
User1	-	0.15	-0.01
User2	0.20	-	0.07
User3	-0.07	0.00	-

Table 6: User network similarity

The randomization process adds a random positive or negative value to the hybrid similarity table values. These values are small enough to disturb the similarity values to have a different nearest neighborhood every time, but capable of producing accurate recommendations. Thus, an attacker will find it more difficult to guess the nearest neighbours of a user.

5 Experimental Evaluation

In this section, we experimentally evaluate, initially the recommendation algorithm and then the privacy model, showing that our integrated method is both practical and effective. The experiments were conducted on a Pentium i3 2.13 GHz with 4GBs of RAM, running Windows 8.1. All of the algorithms have been implemented in Java and extended the Apache Mahout [Owen, Anil, Dunning and Friedman 2011] libraries.

5.1 Evaluation of the recommendation method

We experimentally evaluate our method using the Root Mean Square Error (RMSE) which is a frequently used method that measures the accuracy of recommendation systems [Herlocker, Konstan, Terveen and Riedl 2004] [PampIn, Jerbi and O'Mahony

2015]. Equation 3 shows the Root Mean Square Error, where p_i is the predicted rating and r_i is the actual rating.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - r_i)^2} \quad (3)$$

We use the RMSE measure to compare our approach to the following recommendation approaches to show that it is effective. In RMSE lower values are better.

- Collaborative Filtering

User-based Collaborative filtering is a method where the user ratings are used to provide recommendations. This algorithm has been explained in detail in section 2. In our experiments, we used the algorithm provided by the Apache mahout library in conjunction with the Pearson correlation similarity.

- Trust-Aware Collaborative Filtering

Trust-Aware Collaborative Filtering is an approach where every recommendation is derived from users that belong to the trust network of the user requesting the recommendations.

5.1.1 Real dataset

To evaluate our methods, we used the Epinions dataset, which is a directed dataset of who trusts who in a social rating network. In Epinions everyone can register and provide ratings for products on a scale from 1-5. Moreover, every user can create a trust network of her choice. The network is directed one way only, which means that trust cannot be traced backwards. The dataset has 49 thousand users and 487 edges between them. The dataset contains 140 thousand items with 665 thousand ratings.

5.2 Experiments

The RMSE values for all algorithms based on the Epinions dataset are shown in Figure 2. Figure 3 shows the comparison between CFTrust and CFTrust with privacy and k is the number of neighbours in all cases. Figure 4 shows a Precision and Recall comparison between Collaborative Filtering, CFTrust and CFTrust with privacy, where all the random values have been set between -0.20 to 0.20 for this part of the evaluation.

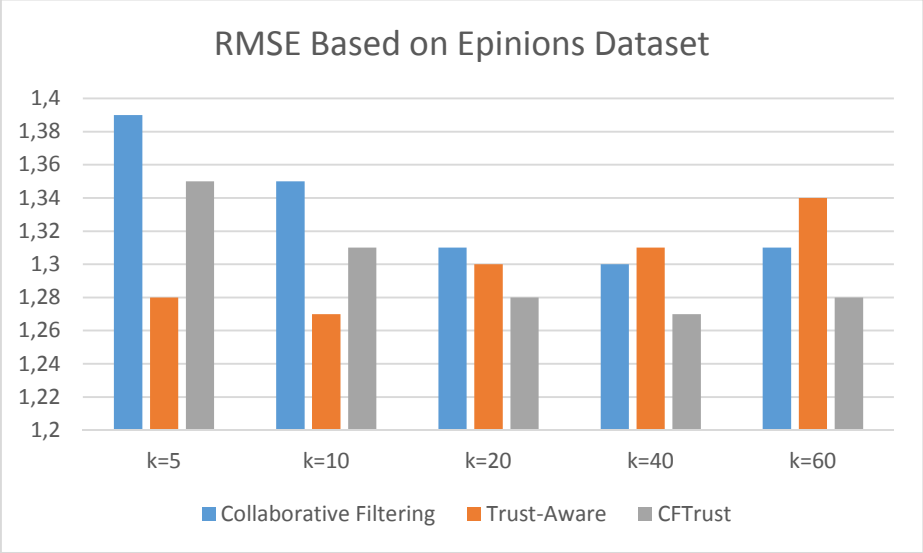

Figure 2: RMSE Comparisons based on Epinions dataset.

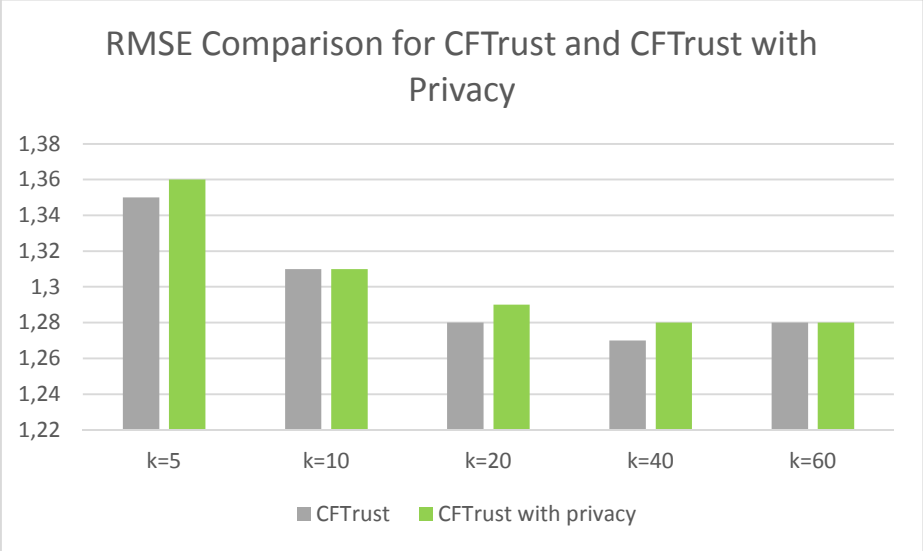

Figure 3: RMSE CFTrust comparisons based on Epinions dataset.

5.2.1 Comparisons

Our proposed method has been compared against existing methods, including the following privacy preservation methods and all the results are shown in Table 7. The comparison is necessary to show the differences in accuracy between different recommendation methods, with and without privacy protection, and among different neighbourhood sizes.

- **Random Perturbations:** This is an approach described in [Berkovsky, Kuflik and Ricci 2012] . It is a method where only a subset of the user ratings is perturbed. An integer variable is defined for every user and before she submits a rating it checks if the current rating submission is to be perturbed. If it is then then a randomly generated integer number, from within a fixed range is added to the rating. The values for this algorithm have been set from -2 to 2 and about 30% of the total ratings of the dataset have been perturbed.
- **Randomized Perturbations:** This is an approach described in [Berkovsky et al. 2012]. It is a method where every rating is perturbed before it is submitted to the server. The values for this algorithm have been set from -1 to 1, while all being integers.

Neighborhood Size	Collaborative Filtering	Collaborative Filtering With Privacy	CFTrust	CFTrust with Privacy	Random Perturbations	Randomized Perturbations
Number of nearest neighbors used for the evaluation	User based collaborative filtering	Applied on neighborhood similarities with values from -0.20 to 0.20	Collaborative filtering combined with the trust network	Collaborative filtering combined with the trust network, plus privacy with values from -0.20 to 0.20	Applied on Collaborative Filtering with values from -2 to 2	Applied on Collaborative Filtering with values from -1 to 1
5	1.39	1.41	1.35	1.36	1.56	1.57
10	1.35	1.36	1.31	1.31	1.52	1.54
20	1.31	1.33	1.28	1.29	1.50	1.51
40	1.30	1.32	1.27	1.28	1.51	1.54
60	1.31	1.32	1.28	1.28	1.52	1.55

Table 7: Comparisons between methods

5.3 Case study

To support our privacy approach, we developed a number of cases, each featuring a different application scenario. We have developed a software application using the Java programming language and the privacy-aware role based access control model. Moreover, the MySQL database, the Apache Shiro and the Apache Mahout libraries were used to make the required tests.

5.3.1 Application scenarios

Scenario #1 Consider an automated software agent *S* that has permissions to the highest privacy level and makes a request to the database to access user data, including location and ratings, to provide personalized recommendations. While *S* not being the owner of the data the system checks, that *S* is an automated agent and not a human and that the purpose the data will be used for is to provide recommendations directly to the user and no further communications with other users at any level will take place. Then the access control policy authorizes *S* to access the data for the required purpose; S being an agent that was registered in the database (where it had the role of software agent). The next step was to verify that S can have access to the data using an access control policy, such as the one in Table 5. Finally, the recommender will gain access to the required database tables, storing the ratings and the trust network, to make the recommendations. Table 5 shows the pseudo code for the access control policy for agent s. The subsequent step is to perform the recommendation step and deliver the recommendations to the user. After the procedure is successfully followed the recommendations for user *Alice*, which we assume is user with id 1 in the Epinions dataset, are shown in Table 8. The recommendations have been delivered using the standard collaborative filtering method with a user neighborhood of 40 and the requested recommendations set at 5.

Access Granted!
RecommendedItem[item:3855, value:5.0]
RecommendedItem[item:980, value:5.0]
RecommendedItem[item:1083, value:5.0]
RecommendedItem[item:9404, value:5.0]
RecommendedItem[item:599, value:5.0]

Table 8: Recommendations for Alice based on Collaborative Filtering

Scenario #2 *Alice* is an end user that wants to use the services offered by the recommendation system. *Alice* however has specified at the settings of the system that she doesn't want to share any data. In this case the access control will take place to restrain access to the user ratings and trust network. Additionally, the final CFTrust table will be altered so the neighbourhood will not be identified from similar recommendations provided by the system. Once again, an access control policy is applied by the system and the neighbourhood randomization technique is applied. Then every time recommendations are produced, different, but similar, neighbors will be assigned to *Alice*. Table 9 is the output of the recommendation algorithm, for

collaborative filtering and CFTrust with and without privacy. The user neighbourhood was set at 40 users for all cases and the number of recommendations was 5. The recommendations are based on the Epinions dataset for user with id 1, assuming it is *Alice*. The results contain the recommendations derived without and with privacy to show that both methods are effective. It is shown that example item 732 remained in the same recommendation place for both approaches, which means that the similar user that this item was retrieved from remained in the same neighbourhood place. While item 891 has stepped up from place 4 to place 1, which means that the similar neighbor scaled to the first position. The results in both cases in tables 8 and 9 offer products that are relevant to the user since the rating prediction for all items is 5, which is the highest predicted value.

CFTrust	CFTrust With Privacy
Access Granted!	Access Granted!
RecommendedItem[item:980, value:5.0]	RecommendedItem[item:891, value:5.0]
RecommendedItem[item:732, value:5.0]	RecommendedItem[item:732, value:5.0]
RecommendedItem[item:676, value:5.0]	RecommendedItem[item:487, value:5.0]
RecommendedItem[item:891, value:5.0]	RecommendedItem[item:6520, value:5.0]
RecommendedItem[item:895, value:5.0]	RecommendedItem[item:676, value:5.0]

Table 9: Recommendations for Alice based on CFTrust

Scenario #3 Consider user *Bob, an employee,* that has permissions required to access level *Medium*, which could mean access to all user data, except ratings. *Bob* wants to access the data of all users for marketing purposes. The access control policy will check that *Bob* is authorized for this action and will release the database tables, generalized to the required level for marketing purposes. Moreover, if *Bob* makes a request to access data in a period which is not within the authorized period, the authorization will not be granted. In this case the system will consult the access control model to see if Bob can have access to the data. An access control policy is applied again at this stage. Then the required data are generalized and delivered to the user. Additionally, data generalization such as the one by Li et al. [22] or Sweeny [23] can be applied. Table 10 shows the record of user *Alice* that *Bob* wants to retrieve for marketing purposes, while Table 11 shows the generalized record of the user. Critical fields such as username, password, name and email have been removed. The address has been generalized to city level only and that the age has been generalized to a group level. Also, the recommendations remained as an important field for marketing purposes.

Id	Username	Password	Name	Address	Age	Top 5 Items
1	Alice22	******	Alice Surname	122 Oxford Street, London, W1 2PK	30	891, 732, 487, 6520, 676

Table 10: Personal Record of user Alice

Id	Address	Age	Top 5 Items
1	London	30-39	891, 732, 487, 6520, 676

Table 11: Generalized Personal Record of user Alice

6 Related Work

The use of collaborative recommendation methods has been heavily utilized by GroupLens, which uses an algorithm that is based on user preferences in order to make predictions for unrated items and make recommendations [Resnick, Iacovou, Suchak, Bergstrom and Riedl 1994]. The algorithm is based on a neighbourhood of most common users and is known as Collaborative Filtering (CF). An improvement of user-based CF is considered to be the item-based approach [Sarwar, Karypis, Konstan and Riedl 2001]. This is based on similarities between items instead of users to make recommendations. Furthermore, there are many methods in the literature that consider user ratings in combination with the trust network of the user to make recommendations. A noticeable example of such a method is TidalTrust [Golbeck 2005] that uses the ratings of a user and then executes a bread-first search algorithm in the trust network in order to make a prediction. MoleTrust [Massa and Avesani 2007] is another method that takes into consideration the trust network up to a user defined path and then makes recommendations based on that information. Another trust-based recommendation has been developed by Jamali & Ester [Jamali and Ester 2009] and is called TrustWalker. This is a method that takes into consideration two things: firstly, it suggests that the social network of the user is an independent source of information and secondly it assumes that strongly trusted friends are more reliable than weakly trusted friends. An inspirational method has been proposed by [Symeonidis et al. 2011] which utilizes the trust network of the user combined with data from multiple social networks. This method also proposes the use of weight variables for the similarity matrices. Furthermore, this method proposes the use of client based personalization when a mobile device is used.

Privacy is an essential factor when it comes to the development of web-based information systems such as recommender systems. Personalized information systems are being developed with privacy-preservation in mind and utilize data from social networks [Massa and Avesani 2007]. An inspirational related work was delivered by [Polat and Du 2005] and shows that even though the ratings can be altered the accuracy of the recommendations is of an acceptable level. This method perturbs every rating before it is submitted to the server. Furthermore privacy is a broad term in recommender systems and can be applied at different stages. [Zhu, Ren, Zhou, Rong and Xiong 2014] have provided an algorithm to protect the nearest neighbourhood from attacks; to do so it uses differential privacy noise addition methods. Other, different approaches and opinions exist in the field of privacy such as ALAMBIC, which was proposed by [Aimeur, Brassard, Fernandez and Mani Onana 2008] and uses a semi-trusted third party, which must be utilized by the server in order for the data to be usable. Data obfuscation techniques have been used by [Parameswaran and Blough 2007] in order to provide privacy-preserved recommendations. [Yakut and Polat 2012] have proposed the arbitrary distribution of data in order to achieve privacy in the recommendations. [Songjie Gong 2011] developed a privacy aware recommender system based on randomized perturbation techniques and secure multiparty computation. A somewhat different approach to privacy is the one offered by [Tada, Kikuchi and Puntheeranurak 2010], where the similarity between items is explored by an adjusted collaborative filtering algorithm. Other works have proposed the use of pseudonyms [Jorns, Quirchmayr and Jung 2007] although its applicability is not convincing to help produce optimal results. [Zhan et al. 2010] proposed a method that is based on cryptology and on the scalar product protocol. Kobsa performed a survey and concluded that the most widely used privacy techniques in recommender systems are pseudonymous users, client side personalization and distribution of data [Kobsa 2007].

Different methods can be used to protect user privacy mainly from the user side. However, our proposed integrated approach utilizes a combination of methods applied at the server side to protect user privacy and preserve accuracy. The aim is for the server to assist its user base by protecting their common data by attacks and unauthorized access, while maintaining high accuracy.

7 Conclusions and Future Work

We have defined a simple similarity function for recommendations derived from friends and friends-of-friends of the trust network of the user requesting the recommendations. In our approach, extra weight was given to privacy preservation, while the accuracy can be maintained. To achieve this, we have adapted and extended a privacy-aware role based access control model. We also introduced the concept of neighbourhood randomization, which gives the ability to have a different user neighbourhood every time recommendations are requested. Our main idea was to keep the recommendation related data, such as the user ratings, restricted from human access and try to keep the user neighbourhood private. This approach provides recommendations without losing any accuracy or with a very small utility cost if the

neighbourhood privacy protection is used. This is quite significant as privacy is a factor that has previously restricted the wider use of recommender systems, while in our approach we have shown that we can maintain high levels of accuracy in the recommendations provided. We have used the collaborative filtering algorithm, which is the most widely used method for providing recommendations. The system has been experimentally evaluated using established measures such as RMSE, Precision and Recall and through the means of a case study of real world application scenarios, while real world data have been used in the measurements. From the experimentation results it is observable that collaborative filtering is becoming more accurate as the user neighbourhood grows. Also, trust-aware is more accurate than the other method when the neighbourhood is small. The combined approach gives better recommendations when the neighbourhood grows larger. When a privacy protection technique is applied (either at the neighbourhood similarities or before a rating is submitted), a decrease in the accuracy is observed. Different privacy-preserving techniques aim to keep a common goal, to preserve privacy, and this can be done at different stages of the recommendation procedure. In the comparisons of our proposed method with the alternatives, we aimed to keep the randomized variables in all the algorithms at sensible levels, for the comparisons to be consistent. It is observable that an alteration of the similarity values that form the nearest neighbourhood of each user, it may preserve privacy while the accuracy loss is considerably smaller than the approaches that alter the ratings before submission to the database of the system. However, a balance needs to be maintained between privacy and accuracy for a system to provide both accurate recommendations and preserve privacy. Our proposed method aims to preserve privacy while maintaining the accuracy at a high level.

The use of collaborative filtering in combination with the trust network gives recommendations of high accuracy. In the future, we aim to investigate the use of other hybridization methods besides and how these can fit with privacy-preserving approaches.

References

[Aimeur, Brassard, Fernandez and Mani Onana 2008] Aimeur, E., Brassard, G., Fernandez, J. M., Mani Onana, F. S.: 'Alambic: A privacy-preserving recommender system for electronic commerce'; International Journal of Information Security, Vol. 7, No. 5 (2008), pp. 307–334. http://doi.org/10.1007/s10207-007-0049-3

[Berkovsky, Kuflik and Ricci 2012] Berkovsky, S., Kuflik, T., Ricci, F.: 'The impact of data obfuscation on the accuracy of collaborative filtering'; Expert Systems with Applications, Vol. 39, No. 5 (2012), pp. 5033–5042. http://doi.org/10.1016/j.eswa.2011.11.037

[Bogers and van den Bosch 2011] Bogers, T., van den Bosch, A.: 'Fusing Recommendations for Social Bookmarking Web Sites'; International Journal of Electronic Commerce, Vol. 15, No. 3 (2011), pp. 31–72. http://doi.org/10.2753/JEC1086-4415150303

[Burke 2002] Burke, R.: 'Hybrid Recommender Systems : Survey and and Experiments. User Modelling and User-Adapted Interaction'; User Modeling and UserAdapted Interaction, Vol. 12 (2002), pp. 331–370. http://doi.org/10.1023/A:1021240730564]

[Carmagnola, Vernero and Grillo 2014] Carmagnola, F., Vernero, F., Grillo, P.: 'Advanced social recommendations with sonars++'; Interacting with Computers, Vol. 26, No. 1 (2014), pp. 75–88. http://doi.org/10.1093/iwc/iwt028

[Chellappa and Sin 2005] Chellappa, R. K., Sin, R. G.: 'Personalization versus privacy: An empirical examination of the online consumer's dilemma'; Information Technology and Management, Vol. 6, No. 2–3 (2005), pp. 181–202. http://doi.org/10.1007/s10799-005-5879-y

[Golbeck 2005] Golbeck, J.: 'Personalizing applications through integration of inferred trust values in semantic web-based social networks'; In CEUR Workshop Proceedings (Vol. 171) (2005), pp. 15–28.

[Goldberg, Roeder, Gupta and Perkins 2001] Goldberg, K., Roeder, T., Gupta, D., Perkins, C.: 'Eigentaste: A Constant Time Collaborative Filtering Algorithm'; Information Retrieval, Vol. 4, No. 2 (2001), pp. 133–151. http://doi.org/10.1023/A:1011419012209

[Herlocker, Konstan, Terveen and Riedl 2004] Herlocker, J. L., Konstan, J. A., Terveen, L. G., Riedl, J. T.: 'Evaluating collaborative filtering recommender systems'; ACM Transactions on Information Systems (TOIS), Vol. 22, No. 1 (2004), pp. 5–53. http://doi.org/10.1145/963770.963772

[Jamali and Ester 2009] Jamali, M., Ester, M.: 'TrustWalker: a random walk model for combining trust-based and item-based recommendation'; Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2009), pp. 397–406. http://doi.org/citeulike-article-id:5151320

[Jannach, Zanker, Felfernig and Friedrich 2010] Jannach, D., Zanker, M., Felfernig, A., Friedrich, G.: 'Recommender systems: An introduction'; Recommender Systems: An Introduction (2010). http://doi.org/10.1017/CBO9780511763113

[Jeckmans et al. 2013] Jeckmans, A. J. P., Beye, M., Erkin, Z., Hartel, P., Lagendijk, R. L., Tang, Q.: 'Privacy in Recommender Systems'; Social Media Retrieval (2013), pp. 263–281. http://doi.org/10.1007/978-1-4471-4555-4_12

[Jorns, Quirchmayr and Jung 2007] Jorns, O., Quirchmayr, G., Jung, O.: 'A privacy enhancing mechanism based on pseudonyms for identity protection in location-based services'; Conferences in Research and Practice in Information Technology Series, Vol. 68 (2007), pp. 133–142.

[Kobsa 2007] Kobsa, A.: 'Privacy-Enhanced Web Personalization'; Communications of the ACM, Vol. 50, No. 8 (2007), pp. 628–670. http://doi.org/10.1145/1278201.1278202

[Massa and Avesani 2007] Massa, P., Avesani, P.: 'Trust-aware recommender systems'; Proceedings of the 2007 ACM Conference on Recommender Systems RecSys 07, Vol. 20 (2007), pp. 17–24. http://doi.org/10.1145/1297231.1297235

[Ni et al. 2010] Ni, Q., Bertino, E., Lobo, J., Brodie, C., Karat, C.-M., Karat, J., Trombeta, A.: 'Privacy-aware role-based access control'; ACM Transactions on Information and System Security, Vol. 13, No. 3 (2010), pp. 1–31. http://doi.org/10.1145/1805974.1805980

[Owen, Anil, Dunning and Friedman 2011] Owen, S., Anil, R., Dunning, T., Friedman, E.: 'Mahout in Action'; Online (2011). http://doi.org/citeulike-article-id:7544201

[PampIn, Jerbi and O'Mahony 2015] PampIn, H. J. C., Jerbi, H., O'Mahony, M. P.: 'Evaluating the Relative Performance of Collaborative Filtering Recommender Systems'; Journal of Universal Computer Science, Vol. 21, No. 13 (2015), pp. 1849–1868.

[Parameswaran and Blough 2007] Parameswaran, R., Blough, D. M.: 'Privacy Preserving Collaborative Filtering Using Data Obfuscation'; In Granular Computing, 2007. GRC 2007.

IEEE International Conference on (2007), p. 380. http://doi.org/10.1109/GrC.2007.133

[Polat and Du 2005] Polat, H., Du, W.: 'Privacy-preserving collaborative filtering'; International Journal of Electronic Commerce, Vol. 9, No. 4 (2005), pp. 9–35. http://doi.org/10.2307/27751163

[Polatidis and Georgiadis 2013] Polatidis, N., Georgiadis, C. K.: 'Recommender Systems: The Importance of Personalization on E-business Environments'; International Journal of E-Entrepreneurship and Innovation, Vol. 4, No. 4 (2013), pp. 32–46. http://doi.org/10.4018/ijeei.2013100103

[Polatidis, Georgiadis, Pimenidis and Mouratidis 2017] Polatidis, N., Georgiadis, C. K., Pimenidis, E., Mouratidis, H.: 'Privacy-preserving collaborative recommendations based on random perturbations'; Expert Systems with Applications, Vol. 71 (2017), pp. 18–25. http://doi.org/10.1016/j.eswa.2016.11.018

[Resnick, Iacovou, Suchak, Bergstrom and Riedl 1994] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: 'GroupLens : An Open Architecture for Collaborative Filtering of Netnews'; Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work (1994), pp. 175–186. http://doi.org/10.1145/192844.192905

[Sandhu, Coyne, Feinstein and Youman 1996] Sandhu, R. S., Coyne, E. J. E., Feinstein, H. L., Youman, C. E. C.: 'Role-based access control models'; Computer, Vol. 29, No. 2 (1996), pp. 38–47. http://doi.org/10.1109/2.485845

[Sarwar, Karypis, Konstan and Riedl 2001] Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: 'Item-based collaborative filtering recommendation algorithms'; Proceedings of the 10th …, Vol. 1 (2001), pp. 285–295. http://doi.org/10.1145/371920.372071

[Shi, Larson and Hanjalic 2014] Shi, Y., Larson, M., Hanjalic, A.: 'Collaborative Filtering beyond the User-Item Matrix : A Survey of the State of the Art and Future Challenges'; ACM Computing Surveys (CSUR), Vol. 47, No. 1 (2014), pp. 1–45. http://doi.org/http://dx.doi.org/10.1145/2556270

[Songjie Gong 2011] Songjie Gong: 'Privacy-preserving Collaborative Filtering based on Randomized Perturbation Techniques and Secure Multiparty Computation'; International Journal of Advancements in Computing Technology, Vol. 3, No. 4 (2011), pp. 89–99. http://doi.org/10.4156/ijact.vol3.issue4.10

[Symeonidis, Tiakas and Manolopoulos 2011] Symeonidis, P., Tiakas, E., Manolopoulos, Y.: 'Product recommendation and rating prediction based on multi-modal social networks'; Proceedings of the 5th ACM Conference on Recommender Systems - RecSys '11 (2011), p. 61. http://doi.org/10.1145/2043932.2043947

[Tada, Kikuchi and Puntheeranurak 2010] Tada, M., Kikuchi, H., Puntheeranurak, S.: 'Privacy-preserving collaborative filtering protocol based on similarity between items'; In Proceedings - International Conference on Advanced Information Networking and Applications, AINA (2010), pp. 573–578. http://doi.org/10.1109/AINA.2010.159

[Tang, Hu and Liu 2013] Tang, J., Hu, X., Liu, H.: 'Social recommendation: a review'; Social Network Analysis and Mining, Vol. 3, No. 4 (2013), pp. 1113–1133. http://doi.org/10.1007/s13278-013-0141-9

[Toch, Wang and Cranor 2012] Toch, E., Wang, Y., Cranor, L. F.: 'Personalization and privacy: A survey of privacy risks and remedies in personalization-based systems'; User Modeling and User-Adapted Interaction, Vol. 22, No. 1–2 (2012), pp. 203–220. http://doi.org/10.1007/s11257-011-9110-z

[Valcarce, Parapar and Barreiro 2015] Valcarce, D., Parapar, J., Barreiro, Á.: 'A Distributed Recommendation Platform for Big Data'; Journal of Universal Computer Science, Vol. 21, No. 13(2015),pp.1810–1829.Retrievedfrom
http://www.jucs.org/jucs_21_13/a_distributed_recommendation_platform

[Yakut and Polat 2012] Yakut, I., Polat, H.: 'Arbitrarily distributed data-based recommendations with privacy'; Data and Knowledge Engineering, Vol. 72 (2012), pp. 239–256. http://doi.org/10.1016/j.datak.2011.11.002

[Zhan et al. 2010] Zhan, J., Hsieh, C. L., Wang, I. C., Hsu, T. S., Liau, C. J., Wang, D. W.: 'Privacy-preserving collaborative recommender systems'; IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews, Vol. 40, No. 4 (2010), pp. 472–476. http://doi.org/10.1109/TSMCC.2010.2040275

[Zhu, Ren, Zhou, Rong and Xiong 2014] Zhu, T., Ren, Y., Zhou, W., Rong, J., Xiong, P.: 'An effective privacy preserving algorithm for neighborhood-based collaborative filtering'; Future Generation Computer Systems, Vol. 36 (2014), pp. 142–155. http://doi.org/10.1016/j.future.2013.07.019