An Extensive Study of Privacy Preserving Recommendation System Using Collaborative Filtering

Abhaya Kumar Sahoo, Chittaranjan Pradhan, Brojo Kishore Mishra, and Bhabani Shankar Prasad Mishra

Abstract Recommender system is the most information filtering-based system that deals with information overload by filtering vital information from large dynamically collected information according to the user's choices, interest, or item's behavior. The collaborative-based filtering recommender system is one of the best filtering approaches, which is very effective in a wide range of applications. The recommender system's accuracy usually depends on the quality of the collected data, which cannot be collected from users without concerning their privacy requirements. Today's recommender systems are obliged to collapse unless they provide a measure of privacy to users. However, privacy and accuracy are conflicting goals because preserving privacy requires a level of distortion in original data, which yields a decrease in recommender systems' accuracy. Nowadays, providing privacy to sensitive information in a side recommender system is the main requirement with the best accuracy. This chapter summarizes all privacy-preserving methods for providing security and privacy to the user's rating of a particular item.

A. K. Sahoo (✉) · C. Pradhan · B. S. P. Mishra
School of Computer Engineering, Kalinga Institute of Industrial Technology,
Bhubaneswar, India
e-mail: abhayakumarsahoo2012@gmail.com

C. Pradhan
e-mail: chitaprakash@gmail.com

B. S. P. Mishra
e-mail: mishra.bsp@gmail.com

B. K. Mishra
Department of Computer Science and Engineering, GIET University, Gunupur, India
e-mail: brojokishoremishra@gmail.com

D. P. Acharjya et al. (eds.), *Deep Learning in Data Analytics*, Studies in Big Data 91,
https://doi.org/10.1007/978-3-030-75855-4_10

1 Introduction

Nowadays, everything is available on the internet. When people buy any product through the internet, they first search for reviews or comments about that product. At that time, people may be confused about whether that product is preferable or not based on comments. So recommendation system provides a platform to recommend such a product, which is valuable and acceptable for people. Such a system is based on the item characteristic, user profiles filled on the website, and products' information. This filtering based system collects a large amount of information dynamically from user's interest, ratings, choices, or item's behavior, filters this information, and provides vital information [1]. Privacy is a major concern in the case of a decision support system. Otherwise, this system wrongly interprets the information. To achieve better accuracy, the collection of relevant data is important. Therefore, different privacy-preserving collaborative filtering approaches are becoming popular with increasing privacy concerns.

The recommender system has the ability to predict whether a particular user would prefer an item or not based on the user's profile. This system can be implemented based on a user's profile or item's profile. This chapter explains different collaborative filtering schemes based recommendation system that provides valuable information to users based on the item's profile. Nowadays, many blog forums are available on different websites where people can give their opinions, reviews, blogs, and comments about the items. After getting ratings about any product by users, the recommendation system makes decisions about users who do not give any ratings [2]. Several e-business websites are taking a recommendation system to increase their revenue in the competitive market. Millions of users buy their products from online e-commerce websites. After buying products, they give their opinions or comments about that product in the respective web forum. So, Generating revenue is the main goal of all entrepreneurs. Using this recommendation system process, we can increase our sales productivity in the market [3]. Security and privacy to recommender systems are required to preserve sensitive information related to the user's rating for a particular item. Different approaches to privacy and security are mentioned in this chapter.

The rest of the chapter is organized as follows: Sect. 2 describes the overview of the recommendation system. Section 3 presents a collaborative, based filtering recommendation system. Section 4 presents security and privacy based collaborative filtering techniques in the recommendation system. Section 5 shows related work and original contributions of the chapter, and Sect. 6 contains the conclusion and future work.

2 Recommendation System Foundations

In the recommendation system, the two main entities play the main role, i.e., users and items. Users give their preferences about certain items, and these preferences must be found out of the collected data. The collected data are represented as a utility matrix that provides each user-item pair's value representing the user's degree of preferences for specific items. In this way, these are mainly two broadcast categories of recommender engine algorithms: user-based and item-based recommenders. In a user-based recommender system, users give their choices and ratings on items. One can recommend that item to the user, which is not rated by that user with a user-based recommender engine, considering similarity among the users. In an item-based recommender system, we use similarity between items (not users) to make predictions from users. Data collection for recommender system is the first job for prediction [1, 4].

2.1 Phases of Recommendation System

The different phases of recommendation system are briefly discussed below.

Information Collection Phase: This phase collects vital information about users and prepares user profile based on the user's attribute, behaviors, or resources accessed by users. Without constructing a well-defined user profile, the recommendation engine cannot work properly. The recommender system is based on inputs collected in different ways, such as explicit feedback, implicit feedback, and hybrid feedback. Explicit feedback takes input given by users according to their interest in an item, where implicit feedback indirectly takes user preferences by observing user behavior. Hybrid feedback can be collected as both explicit and implicit feedback [1].

Learning Phase: This phase takes feedback gathered in the information collection phase as input and processes this feedback using a learning algorithm and exploits the user's features as output [1].

Recommendation Phase: Preferable items are recommended for users in this phase. By analyzing feedback collected in the information collection phase, prediction can be made, which is happening through model or memory-based or observed activities of users by the system [1].

3 Filtering Techniques Based Recommendation System

An efficient recommendation technique is essential to provide a useful recommendation to its individual users. This explains three types of recommendation techniques, mainly used to provide recommendations to users about the item. The following Fig. 1 shows the hierarchy of a recommender system based on different filtering techniques [2].

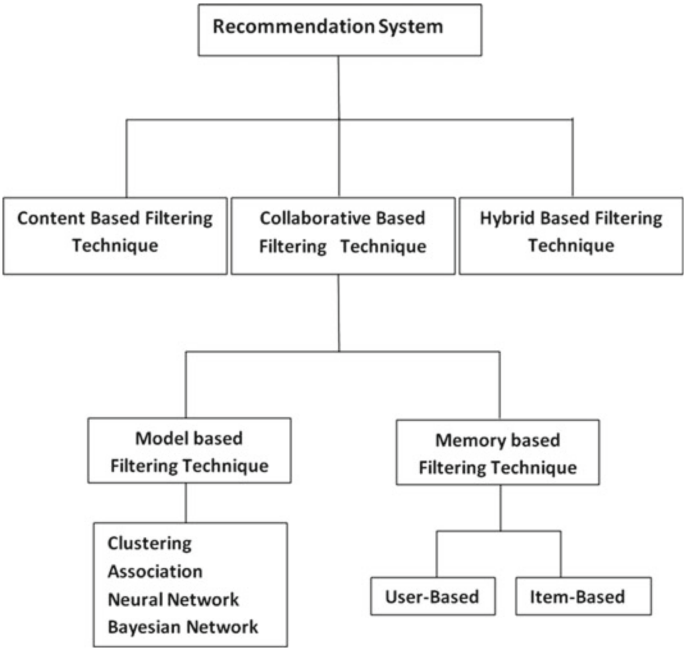
Recommendation System
Content Based Filtering Technique
Collaborative Based Filtering Technique
Hybrid Based Filtering Technique
Model based Filtering Technique
Memory based Filtering Technique
Clustering
Association
Neural Network
Bayesian Network
User-Based
Item-Based

Fig. 1 Hierarchy of recommender system based on filtering

3.1 Content Based Filtering Recommendation System

The content-based filtering technique focuses on the analysis of features and attributes of items to generate predictions. Content-based filtering is usually used in case of document recommendation. In this technique, the recommendation is based on user profiles, which deal with different items and the user's previous buying history. Users give their preferences in terms of ratings, which are positive or negative, or neutral in nature. In this technique, positively rated items are recommended to the user [1, 2].

3.2 Collaborative Filtering Recommendation System

Instead of considering the features and attributes of items to determine their similarity, collaborative filtering (CF) this approach uses user-based ratings to find similarity between items. After collecting all user ratings, the system compares these ratings with other users with a utility matrix and recommended items. We use different distance measures to approach Jaccard's distance, cosine distance, and Pearson's coefficient to find a user's similarity. This filtering method is usually used in an e-

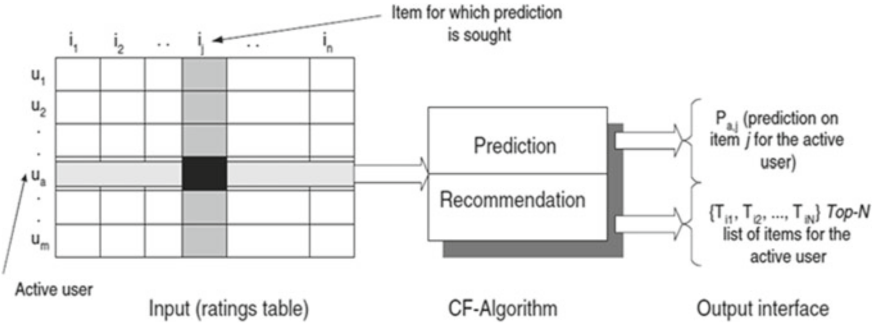
Item for which prediction is sought
i1
i2
ij
in
u1
u2
ua
um
Active user
Input (ratings table)
Prediction
Recommendation
CF-Algorithm
Pa,j (prediction on item j for the active user)
{Ti1, Ti2, ..., TiN} Top-N list of items for the active user
Output interface

Fig. 2 Collaborative filtering technique

commerce website to recommend items based on users' ratings [5]. Collaborative filtering predicts unknown outcomes by creating a user-item matrix of choices or preferences for items by users. Similarities between users' profiles are measured by matching the user-item matrix with users 'preferences and interests. The neighborhood is made among groups of users. The user who has not rated specific items before gets recommendations to those items by considering positive ratings given by users in his neighborhood. The CF in the recommendation system can be used either in prediction or recommendation. Prediction is a rating value $R_{i,j}$ of item j for user i. This collaborative filtering technique is mainly categorized in two directions: memory-based and model-based collaborative filtering. The following Fig. 2 explains the whole process of collaborative filtering technique [6, 7].

Memory Based Collaborative Filtering: Item and user are two key factors in this filtering technique. So this technique comprises two ways, such as item-based collaborative filtering and user-based collaborative filtering. Prediction is calculated by measuring similarity among the items [6]. This technique builds a model on item similarities by considering all items rated by an active user from the user-item matrix, by which we can measure the similarity among the target item and all retrieved items. Then we select k most similar items, and prediction is calculated by considering a weighted average of the active user rating on similar items k. Different mathematical methods are used to compute similarity among item and user. These are correlation-based similarity measures, cosine-based similarity measures, and Pearson's correlation coefficient.

The Pearson's coefficient is defined in Eq. 1, where $S(a, u)$ represents similarity between two users a and u. $r_{a,i}$ and $r_{u,i}$ denote rating on an item i by user a and u respectively, whereas $\bar{r}_a$ and $\bar{r}_u$ are mean rating given by user a and u respectively, while n is the total number of items in user-item matrix.

$$S(a,u) = \frac{\sum_{i=1}^{n}(r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i=1}^{n}(r_{a,i} - \bar{r}_a)^2}\sqrt{\sum_{i=1}^{n}(r_{u,i} - \bar{r}_u)^2}} \tag{1}$$

Cosine similarity is defined in Eq. 2, is a vector space model which is based on linear algebra. This method measures similarity between two n-dimensional vectors based on angle between them. It is mainly used in information retrieval and text mining. The similarity between two items u and v can be denoted as:

$$S(u, v) = \frac{\sum_{i=1}^{n} r_{u,i}\ r_{v,i}}{\sqrt{\sum_{i=1}^{n} r_{u,i}^2}\sqrt{\sum_{i=1}^{n} r_{v,i}^2}} \tag{2}$$

Jaccard similarity of sets A and B is the ratio of the size of the intersection of A and B to the size of their union. It is defined in Eq. 3.

$$JS(A, B) = \frac{A \cap B}{A \cup B} \tag{3}$$

In the user-based collaborative filtering technique, users' similarity is measured by comparing ratings on the same item. The recommender system predicts rating for an item by the active user by calculating a weighted average of item ratings by users. In this way, the above three methods are used to measure the similarity between two items.

Model Based Collaborative Filtering: This approach is based on previous ratings to learn a model that uses machine learning or data mining techniques. Different approaches, like association rules, clustering, decision tree, artificial neural network, regression, Bayesian classifiers are used to classify user and item based on model [1].

Association Rule Mining: Association rule mining algorithms generate association rules which decide the relationship among items in a transaction. Association rule $A \rightarrow B$ means item set A predicts item set B. This rule can be fitted to a recommendation model to predict the user, and item [8].

Clustering: Clustering is a method used to partition a set of data into number of clusters. A good clustering method means high intra-cluster similarity and low inter-cluster similarity. In the recommendation system, users can partially participate in different clusters, and the degree of participation can be calculated by taking the average across the clusters of participation [9].

Decision Tree: Decision tree makes a graph like a tree structure constructed by considering the training data set in which class labels are known. This tree can be used to classify test data. The decision tree is one type of classifier which handles and classifies previous unseen examples.

Artificial Neural Network: Artificial neural network (ANN) is a network of many connected neurons arranged in different layers. The weight and bias factors are associated with every neuron of each layer. Each neuron has a transfer function through which it measures input, processes this input, and gives the output. This ANN is a classification technique to classify test data.

Regression: Regression is an analysis method where two or more variables are related to each other. One variable is dependent, whereas one or more are independent variables. This regression technique comprises prediction, curve fitting, and hypothesis testing, which create relationships among variables.

Bayesian Classifier: Bayesian classifier is used to solve classification problem based on conditional probability and Bayes theorem. The Bayesian classifier predicts the class by considering the class's probability concerning particular attributes by applying Bayes' theorem. This classifier is usually useful when users' preferences change concerning the time required to build the model.

3.3 Hybrid Filtering Recommendation System

This technique comprises the above two methods to increase the accuracy and performance of the recommendation system. The hybrid filtering technique can be achieved using any of the following ways: building a unified recommendation system that combines both above two approaches, applying collaborative filtering in the content-based approach, and utilizing some content-based filtering in the collaborative approach. This technique uses different hybrid methods such as cascade hybrid, weighted hybrid, mixed hybrid, and switching hybrid according to their operations [1].

4 Security and Privacy Based Collaborative Filtering Techniques in Recommendation System

There are different privacy-preserving techniques used in collaborative filtering based recommender system to provide security to sensitive information related to the user and item [10].

Non-negative matrix factorization (NMF): The randomization method's basic principle is to perturb the data so that the server can only know its range. The server should not know each user's true ratings to a particular item in the recommender system with m users and n items. The NMF method provides privacy to the user's rating of items. The non-negative matrix factorization method is a mathematical method in which multivariate analysis and linear algebra can be used. An original matrix V is factorized into two matrices W and H, with the property that all three matrices have no negative elements [11]. It is defined in Eq. 4, where v_i is the ith column vector of the product matrix V and h_i is the i^{th} column vector of the matrix H.

$$v_i = Wh_i \tag{4}$$

When multiplying two matrices, the dimensions of the factor matrices should be lower than those of the product matrix, and it is the property that forms the basis of NMF. NMF generates factors with significantly reduced dimensions compared to the original matrix. For example, if V is an $(m \times n)$ matrix, W is an $(m \times p)$ matrix, and H is a $(p \times n)$ matrix, then p can be significantly less than both m and n. Finally, each user sends these values to the server where the user-item matrix R is created. The matrix R is factorized into W and H using NMF, the rating of user i for item j is defined in Eq. 5, where r_i and σ_i are mean value and standard deviation of user i respectively; $sgn(R : (i, j))$ is the signature of element in the row i and column j of the R'.

$$P_{i,j} = \bar{r}_i + sgn(R : (i, j)).\sigma_i.[W(i).H(j)] \tag{5}$$

Privacy preserving naive Bayesian classification (PPNBC): PPNBC have proposed to offer NBC-based predictions where each user u_i divides her items into M groups perturbs the ratings of every group n individually [12]. It provides accurate referrals with privacy; efficiency significantly decreases. Besides, it has suggested that online performance and even accuracy of PPNBC can be improved by preprocessing disguised data off-line. There are two preprocessing methods used to perturb ratings. The first method, where neighborhood formation is obtained, determines the best similar items to each item using a binary similarity measure. In the second method, some of the randomly chosen empty cells are filled in the disguised user-item matrix with personalized ratings estimated by PPNBC. Neighborhood formation is used to improve online performance by decreasing the amount of data involved in the collaborative filtering process. For each item in collection data set D, a neighborhood is formed by selecting the best similar items. The number of items in a neighborhood directly affects efficiency. Similarly, an item's neighbors significantly influence preciseness. The best N similar items are chosen as neighbors, and the optimum value of N can be determined experimentally, where N is a constant with $N < m, m$ is the total number of columns in the user-rating matrix. To determine the neighbors by providing the best results, a binary similarity measure should be used. If more importance to commonly rated items is given, the Tanimoto coefficient is preferable. Let $S_{i,j}$ be the number of occurrences of commonly rated items with i in the first pattern and j in the second pattern, where $i, j \in 0, 1$. Given two binary feature vectors X and Y, $T(X, Y)$ denotes modified Tanimoto coefficient between X and Y and it can be calculated using Eq. 6

$$T(X, Y) = \frac{(s_{1,1} + s_{0,0}) - (s_{1,0} + s_{0,1})}{s_{1,1} + s_{0,0} + s_{1,0} + s_{0,1}} \tag{6}$$

It is to be noted that, $S_{1,1}$ is the number of users who have rated both items as 1. $S_{1,0}$ represents the number of users who rated item i like 1 and item j like 0. $S_{0,1}$ is the number of users rated item i as 0 and item j as 1 and $S_{0,0}$ shows the number of users rated both items as 0. Tanimoto similarity measure computes the similarity between two binary vectors. Due to underlying data perturbation methods, it becomes challenging to estimate the same similarities from perturbed data.

Homomorphic encryption (HE): Let $U = \{u_1, u_2, \ldots, u_n\}$ be a set of users, where n is the number of users. Let $I = \{i_1, i_2, \ldots, i_m\}$ be a set of items, where m is the number of items. Let $r_{i,j}$ be a rating given by user u_i for item j, for $i = 1, 2, \ldots, n$, and $j = 1, 2, \ldots, m$. Users do not evaluate all of the items. We denote a missing rating by $r_{i,j} = \Phi$. We assume that the matrix of rating contains many missing elements, that is, a sparse matrix. CF's goal is to predict a missing rating based on the other users' preference for the given item. This model providing privacy by using the homomorphic encryption method gives that users are willing to get recommendations for items that they are not seen before, but at the same time, they are concerned about the privacy of rating made by themselves [13]. We use a public-key cryptosystem E to preserve users' privacy, satisfying an additive homomorphic property by taking M_1, M_2 messages.

$$E[M_1]E[M_2] = E[M_1 + M_2] \tag{7}$$

$$E[M_1]^{M_2} = E[M_1 M_2] \tag{8}$$

Paillier cryptosystem and ElGamal cryptosystem provide key generation and decryption processes distributed among semi-trusted authorities by sharing private keys. The Paillier cryptosystem consists of three steps, i.e., key generation, encryption, and decryption.

1. *Key generation:* Let n be pq, multiplication of large primes p and q, $g \in Z^*_{n^2}$ be a generator whose order divides n. Compute $\lambda = LCM(p-1, q-1)$ and $\mu = (L(g^\lambda \mod n^2))^1 \mod n$, where L is defined by $L(u) = (u1)/n$. The public key is (n, g), and the private key is (λ, μ).
2. *Encryption:* The encryption can work for any m in the range $0 \leq m < n$ with a random number $r \in (0, n)$. Compute ciphertext $c = g^m.r^n \mod n^2$.
3. *Decryption:* Given ciphertext $c \in (0, n^2)$, plaintext m is computed by $m = L(c^\lambda \mod n^2).\mu \mod n$.

The basic idea of the above method comprises two parts, i.e., precomputation phase and recommendation phase. In the precomputation phase, a single user for all ratings is made by other uses. The recommendation phase allows any user to predict any missing rating by using public similarities between items.

DWT based privacy-preserving: The DWT-based CF approach divides the original user-item matrix into two components, i.e., called approximation and detail coefficient. The perturb of individual users' data is discussed in [14].

$$C_{apprx} = \frac{u_j + u_{j+1}}{\sqrt{2}} \tag{9}$$

$$C_{dtl} = \frac{u_j - u_{j+1}}{\sqrt{2}} \tag{10}$$

It is to be noted that, C_{appx} denotes approximation and C_{dtl} denotes detail coefficient. With privacy as a concern, there are two aspects of preserving individuals' privacy in a recommender system, i.e., preventing the server or the data collector from learning the true preferences of users and items, which are actually rated by users. In other words, users want to protect their true ratings about various items and hide their rated and/or unrated items. To achieve the first goal, users generate random numbers and add them to their votes so that the server cannot learn true ratings. They also insert random numbers into some of their unrated item cells to mask their rated and/or unrated items. In order to generate random numbers, users can either use a uniform or Gaussian distribution. In uniform distribution, they generate uniform random values in a range $[-\alpha, \alpha]$, where α is a constant.

Random numbers are generated using normal distribution having to mean as 0 and the standard deviation in Gaussian distribution. After generating random numbers, all users disguise their private ratings with random numbers by adding them to the true votes. Then all users mask their rated and/or unrated items by inserting random values into some of their unrated item cells before sending their vectors to the server. To offer DWT-based recommendations, the server needs normalized ratings. Thus, before perturbing their data, users first normalize their votes by transforming them into z-scores. This transformation process helps users disguise their private data with random numbers generated from a narrower interval. A privacy-preserving scheme that helps to produce accurate predictions based on DWT efficiently without exposing user's privacy is proposed [12]. It has also recommended methods to order items before applying DWT to boost accuracy.

Random perturbation and anonymization: Traditional anonymization scheme, which is based on user's data in the central storage, i.e., is introduced [15]. In this case, any one of the collaborative users is not allowed to access other's data. Each user has to anonymize his or her data individually. The traditional anonymization scheme cannot be useful for distributed anonymization. Further distributed anonymization scheme is proposed that is written as below.

1. The authorized user u_a selects two secrete large primes D_1 and D_2 and then computes $M = D_1 \times D_2$, $|U|$ and $|O| << M$. $e_a = \Phi(M) - d_a$, where D_1, D_2 and d_a are kept by u_a as the private parameter. Then, the public parameters e_a and M are delivered to a group of collaborative users $\{u_c\}$. Note that the selection of the above parameters follows the principle of RSA.
2. For anonymization of data profile, u_c needs to generate some pseudonyms and then assigns the item to these pseudonyms. The anonymized bipartite graph G_c^* is generated with e_a and $M.G_c^*$ can be obtained by e_a iterations of the anonymization map T:
$$(c^*, o^*) = T^{e_a}(\hat{c}, o) \mod M \tag{11}$$

   It is to be noted that $\hat{c} = c + d_c \mod M$, $0 \leq c^* < M$ and $0, 0^* \in [0, M]$. c^* is a pseudonym and the item o^* is assigned to the pseudonym c^*. d_c is the private parameter of u_c. In Eq. 11, the edge (c, o) in G_c is mapped to the edge (c^*, o^*) in G_c^*.

3. When u_a collects all the anonymized bipartite review graphs G_c^* from U_{co}, u_a can obtain the information about R_c from G_c^* with the private parameter d_a as follow:

$$T^{d_a}(c^*, o^*) \mod M = T^{d_a}(T^{e_a}(\hat{c}, o)) \mod M \tag{12}$$

where $\Phi(M)$ is the period of anonymization map T. $\hat{c}$ pseudonym of u_c.

It is not a problem for u_a to recover the information about the structure and edge label of G_c when $\hat{c}$ is used. After obtaining all R_c, $c \in U_{co}$, u_a can use collaborative filtering to predict the ratings of the his/ her interested items. If the unauthorized users u_h tries to learn the private information from e_a and M, they need to face the hard problem of integer factorization. Moreever, due to adopting the multiply pseudonyms and the fake edges, G_c^* can not reveal the information R_c, about the structure and the edge label of G_c by the structure-based and label-based attacks.

Belief propagation (BP): . Belief propagation method [16] uses factor graph. A factor graph is a bipartite graph representing the factorization structure of a global function of many variables, where variable nodes and factor nodes represent variables and local functions, respectively. An edge connects a variable node to a factor node if and only if the variable is an argument of the local function represented by the factor node. The sum-product BP algorithm is a message-passing algorithm that operates on the factor graph, in which messages are exchanged between the factor nodes and variable nodes.

In standard belief propagation, an undirected graphical model with vertex set X is drawn for which the underlying network is a tree structure. This is denoted by $V(X_i)$, the set of possible values of $X_i \in X$. For each $X_i \in X$, we are given a non-negative potential function ψ_i over possible values $x_i \in V(X_i)$. Similarly, for each pair of adjacent vertices X_i and X_j, we are given a non-negative potential function $\Psi_{i,j}$ over joint assignments to X_i and X_j. The main inductive phase of the belief propagation algorithm is the message-passing phase. At each step, a node X_i computes a message $\mu_i \to j$ to send to some $X_j \in N(X_i)$. This message is indexed by all possible assignments $x_j \in (Xj)$.

Belief propagation follows message-passing protocol, in which any vertex of degree d that has received the incoming messages from any $(d1)$ of its neighbors can perform the computation above to send an outgoing message to its remaining neighbor. Eventually, the vertex will receive a message back from this last neighbor, at which point it will be able to calculate messages to send to its remaining $(d1)$ neighbors. The protocol begins at the leaves of the tree, and it follows from standard arguments that until all nodes have received incoming messages from all of their neighbors, there must be some vertex that is ready to compute and send a new message. The message-passing phase ends when all vertices have received messages from all of their neighbors.

Variable weighted BSVD method: Singular value decomposition (SVD) method is a common matrix decomposition technique [13]. It can extract the algebraic feature efficiently, and it also can reveal the matrix structure. SVD can be used in the CF to find the feature of the user and item. In the SVD-based privacy-preserving scheme, the data is arranged as a matrix. Supposed that the original data includes n users and m rating items, then the data can compose an $(n \times m)$ matrix A. The row of the matrix represents the ratings of all the items given by one user, and the column represents the rating of one item given by all users. The $(n \times m)$ matrix A is decomposed into three matrices by SVD.

$$A_{n\times m} = U_{n\times r} \times \sum_{n\times r} \times V^T_{m\times r} \tag{13}$$

Here, U is an $(n \times r)$ matrix, and it represents n users and r subjects. It is the similarity matrix, and it represents the similarity between the user and the subject. The diagonal data are equal to 0, the diagonal data are sorted from the lowest to the highest, and the value is called the singular value. The rank of the matrix is denoted as r, and the value on the diagonal represents the subject strength. V is an $(m \times r)$ matrix, and it is the similarity matrix between the subjects. The nonzero value is only on the diagonal. The SVD of the matrix A can be described as truncated SVD. There is a parameter k in the basic SVD algorithm, where $0 \leq k \leq r$. The disturbing data is described in Eq. 14, where U_k is an $(n \times k)$ matrix which is composed by the first k column of the matrix U.

$$A_k = U_k \sum_k V_k^T \tag{14}$$

In the BSVD algorithm, the larger the k is, the less disturbed the value is, and the data has better usability, but the security is more descendant. The BSVD disturbs the data in the same intensity. In some application scenarios, there are many kinds of definitions about privacy, and some people have cared for their privacy, but some people only concern about the privacy of sensitive data and do not care for the common data. The data must be disturbed in different intensities. A variable weighted BSVD privacy-preserving scheme is presented in the literature [13]. The variable weight means that the users can choose the weight of the privacy-preserving according to their needs. The users can choose to disturb the data if they concern about their privacy, and they also can choose not to disturb the data if they do not care about the privacy disclosure. If the users want to hide that what they have rated and what they did not, they can disturb the data they did not rate. The data users receive the various disturbing data, and they cannot be certain which items are disturbed and which are not disturbed, and they cannot decide which items are rated by certain people and which are not. The variable weight can be calculated using Eq. 15.

$$\omega_i = e^{\delta_{max} - \delta_i} \tag{15}$$

It is to be noted that σ_i represents the disturb weight of user i, $\sigma_m ax$ represents the maximum disturb weight. ω_i represents the variable weight of user i. The variable weight is a monotone increasing function. The higher demand of privacy preserving, the larger is the function value. The range of the function is [0, 1]. The new A_k can be calculated using Eq. 16.

$$A_k = A V_k V_k^T \tag{16}$$

The A_k represents the disturbing matrix, the row represents the sample data, and the column represents the items. A is the original data matrix. V_k is an $n \times k$ matrix, and the column is the feature vectors according to the largest k eigenvalue of matrix C. V_k^T is the transposition of matrix V_k. The C_ω can be calculated using Eq. 17.

$$c_\omega = \sum_{i=1}^{m} \omega_i x_i^T x_i \tag{17}$$

It is to be noted that, x_i represents the rating of user i, and the x_i^T represents the transposition of x_i. The algorithm WSBSVD has the following steps.

1. The original data matrix A is decomposed by the SVD algorithm, and three matrices U, $\sum$ and V are obtained.
2. Select the parameter k, and get the disturbed matrix A_k.
3. Calculate the matrix C_ω and get the new A_k.
4. Use the Pearson similarity measure formula to calculate the similarity matrix SIM.
5. Then calculate prediction.

Randomization method: Randomized method is proposed by Z. Batmaza et al. [17]. In this section, privacy-preserving frameworks and their pseudocodes are represented. The frameworks are designed in such a way so that the appropriate one can be chosen based on the user requirements. The frameworks are structured based on the following three dimensions.

1. Randomization Type: User preferences can be represented using numeric or binary ratings. In numeric ratings based CF schemes, data is masked using randomized perturbation techniques (RPT). On the other hand, data is perturbed using randomized response techniques (RRT) in binary ratings-based CF systems. Thus, the frameworks can be grouped into two classes according to the randomization type as RPTs and RRTs.
2. Confidential Data: Users concern about their private data. Public data can be shared with e-commerce sites. However, users do not want to provide their private data in the plain form to online vendors. Generally speaking, confidential data can be grouped into two broad categories, as ratings and rated or unrated items.
   Ratings: Users usually do not want others to learn their true ratings about the products they bought or showed interest in. Such ratings can be exploited for profiling, unsolicited marketing, price discrimination, government surveillance, and so on. Due to these privacy risks, users mask their ratings and send perturbed ratings to the CF system.
   Rated/unrated items: In addition to ratings, it might be more damaging to reveal rated/unrated items. Rated items disclose information about users' tendencies and interests. Nobody wants others to learn that she bought specific magazines or

products. Likewise, unrated items mean that users have not bought them yet; thus, such products are prospective items to be purchased, and the users are potential customers. This might lead to unsolicited marketing and price discrimination. Due to these reasons, users mask their ratings and rated/unrated items.

3. Privacy Concerns: Privacy concerns differ from one user to another. Varying privacy concerns cause variable data masking. On the other hand, users can perturb their confidential data in the same way, known as invariable data masking. The frameworks can be grouped into two classes based on privacy concerns dimension as invariable and variable.
   Invariable: In this case, all users disguise their private data consistently. The CF system forces each user to mask their private data following the same procedure. Invariable data disguising is less complex and provides a smaller privacy level.
   Variable: Some users are privacy fundamentalist, while some are marginally concerned about their privacy. The majority of the users are less concerned about their privacy than the fundamentals. Due to varying privacy concerns and expectations, users might decide to mask their data inconsistently.

Our frameworks are based on RPTs and RRTs. We first basically describe these methods. RPTs select a random number distribution with zero mean and a standard deviation. Uniform or Gaussian random number distributions can be employed to generate random noise. Using the distribution with the chosen parameter values, the required numbers of random numbers are generated. These random values are finally added to the private data items that are supposed to be disguised. To mask rated/unrated items, randomly selected some of the unrated item cells are filled in with random numbers, too. All users can use the same random number distribution with the fixed values of privacy control parameters. For variable disguising, each user can randomly select random number distribution and the parameters' values over a specified range. RRTs first determine the number of groups and a threshold. If the ratings are grouped into a single group, this scheme is known as the one-group method. If the number of groups two or more, then the corresponding method is called a multi-group scheme. To mask true binary ratings, the ratings are grouped according to the number of groups. Data in each group is then independently masked. A random number is chosen for each group, and the random numbers are compared with the chosen threshold. If the random number is larger than the threshold, binary ratings are reversed (1s are transformed into 0s and 0s are transformed into 1s) and sent to the CF system. Otherwise, the true ratings are sent. To perturb rated/unrated items, randomly selected some of the unrated item cells are filled in with binary ratings. For both RRTs and RPTs, several randomly selected unrated item cells depend on the amount of the unrated item cells. However, the number of unrated cells to be filled in should be carefully determined. Filling too many unrated item cells might significantly affect the originality of the collected true data. Hence, it is reasonable to determine the number of unrated item cells to be filled in based on the user ratings vector density. Note that users must divide their rating vectors into the same number of groups for estimating recommendations with decent accuracy in RRTs.

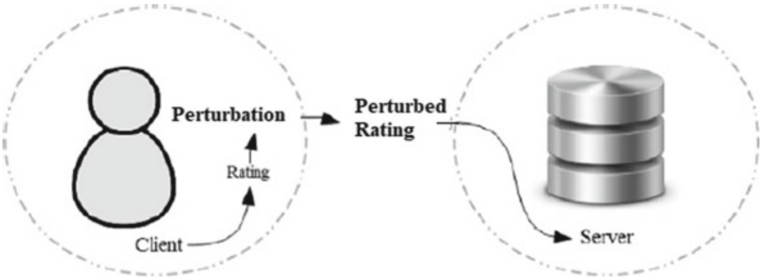

Fig. 3 Privacy preserving rating submissions

Johnson–Lindenstrauss Transformation (JLT): M. Yang and T. Zhu have explained about differential privacy, which is a provable privacy notation [18]. It provides a strong privacy guarantee that the queries' outputs on the neighboring datasets will be statistically similar. The formal definition of differential privacy is presented as follows: A randomized algorithm M gives differential privacy for any pair of neighboring datasets D and D', and for every set of outcomes O, M satisfies:

$$P_r[M(D) \in O] \leq exp(\varepsilon).P_r[M(D') \in O] \tag{18}$$

ε is the privacy parameter, which is defined as the privacy budget. It controls the privacy preservation level. A smaller ε represents greater privacy. The Johnson–Lindenstrauss transform projects the points in a high dimension to a lower-dimensional space while the Euclidean distances between any pair of points are preserved.

The Johnson–Lindenstrauss lemma states that a random matrix X can project the dataset from d dimension to m while maintaining the relative distance. The calculation of similarity between users has a linear relationship with l_2 distance. Therefore, the similarity also can be guaranteed after transformation.

Modified random perturbation (MRP): N. Polatidis et al. have proposed a centralized approach based on rating privacy [10]. This method is focused on the personalization of privacy. In this method, the perturbation range is created randomly for each submitted rating. So, each perturbed rating explores no information about the actual perturbation range that has been used. It utilizes a random value to the actual rating before it is submitted to the server. Randomization is a widely-used privacy preservation method in privacy-preserving data mining. Furthermore, the benefit of using a simple logic algorithm for perturbation is that it can be easily installed on the client-side and that it is less complex to use and understand. A high-level overview of this approach is shown in Fig. 3.

In this method, the insertion of multiple privacy levels (with each level perturbing the rating with a different range of values) protects user privacy. The values of 'minrating' and 'maxrating' refer to the rating scale used by the recommender system. For example, if we have a scale in the range 1–5, then minrating 1 and maxrating refer to 5. If a value, due to the perturbation method, drops below minrating, then the perturbed rating takes the value of minrating, and in the case that it exceeds maxrating, then the perturbed rating takes the value of maxrating.

5 Related Work and Original Contribution

As CF has become more complex, different survey papers have been published on the privacy and security of the CF-based recommender system. Initially, introduction of CF's core concepts: the theory and practice, the rating systems and their acquisition, evaluation, interaction interfaces, and privacy issues are presented. Further, CF filtering methods and compare their results are studied. A survey of CF techniques are also studied. Authors introduce CF's theory and concisely deal with the main challenges: sparsity, scalability, synonymy, gray sheep, shilling attacks, privacy. They also expose an overview table of CF techniques. This chapter focuses on explaining carefully how the most used algorithms in RS work. The chapter also presents CF's basic concepts and evaluation metrics, different security and privacy approaches. Here we summarized all the privacy-preserving RS algorithms in Tables 1 and 2.

6 Experimental Result

Here we compared all existing methods using the MovieLens dataset for finding the mean absolute error (MAE) by taking two distance measures such as Cosine coefficient and Pearson correlation and neighborhood size k, i.e., 20 is equal to 20. Lesser is the MAE value; more is the accuracy. From the experimental result, we analyzed that the coined coefficient distance measure method provides more accuracy than the Pearson correlation method in different privacy-preserving techniques used in the recommendation system. Among all the methods, the random perturbation method gives less MSE value with preserving privacy [10]. It is presented in Fig. 4.

7 Conclusion

An item-based collaborative filtering recommendation system is better to use when users are far greater than the number of items. This system's performance can be affected by data sparsity, cold start problems, shilling attacks, and privacy. So there is a great chance of the future research area. Even though collaborative recommender systems have matured as a concept, have found numerous practical applications in the business world, and inspired a good research volume in academia, there are still privacy concerns from users of such systems. Users want useful recommendations,

Table 1 Comparision among different methods in security and privacy in CF based recommendation system

Privacy preserving RS algorithm	Contribution
Non-negative matrix factorization method with random perturbation technique	1. The basic idea of randomization is to perturb the data in such a way that the server can only know the range of the data and such range is broad enough to preserve users' privacy 2. MAE measures the prediction veracity by calculating the difference between the predicted ratings and the real ratings, the smaller MAE, the better the recommendation quality 3. The Gaussian distribution results are a little better then the uniform distribution ones. The prediction accuracy of this algorithm decreases, but as compromising the privacy, the performance should be acceptable 4. MAE = 0.7 for 100 users
Naive Bayesian Classification method with random perturbation technique	1. Due to its given importance to commonly rated items, Tanimoto coefficient is preferable 2. Two preprocessing methods such as neighborhood formation and increasing density to perturbed ratings for better privacy and enhancement of accuracy
Homomorphic Encryption method	1. The proposed scheme is secure under the security of the cryptographical primitives including additive homomorphic public-key algorithm and the assumption of trusted authorities in case of item-based collaboration filtering 2. MAE = 0.758 for 100 users
DWT Based privacy-preserving method using perturbation technique	1. This approach basically divides the original user-item matrix into two components for each pair called approximation and detail coefficients 2. DWT-based CF scheme can be applied to naturally ordered items and this scheme can overcome the scalability problem by providing high privacy and accuracy 3. MAE = 0.835 for 125 users
Random Perturbation and Distributed Anonymization Scheme	1. This algorithm allows users individually to anonymize their own data without accessing each other's data 2. Anonymization can be achieved by the anonymized map and this scheme can preserve the privacy of collaborative users and outperform the perturbation-based scheme

but at the same time are anxious about submitting ratings due to privacy concerns, thus leading to poor recommendations. One of the most successful means of protecting user privacy in collaborative filtering systems is to perturb the rating before it is submitted to the server. Our proposed method has used a multi-level perturbation method that perturbs each rating at the client-side before it is submitted to the server. Experimental results demonstrate that our proposed method provides acceptable outcomes in terms of accuracy, SSE, and VD values compared to a similar alternative. Furthermore, different privacy protection measures provide different accuracy results, SSE, and VD values, thus offering different protection levels. This is because

Table 2 Continuation of Table 1

Privacy preserving RS algorithm	Contribution
Semi-distributed Belief Propagation method	1. This algorithm computes item similarity as a probabilistic inference problem on the factor graph 2. To avoid disclosing user ratings to the server or other user peers, semi-distributed architecture is used for providing high accuracy with privacy 3. MAE = 0.815 for k = 10 (parameter)
Variable weighted BSVD method	1. The users can disturb their original data with different weights according to their needs 2. The improved slope one algorithm is used to get the prediction 3. MAE = 0.735 for k = 20 (parameter)
Randomization Method	1. Randomization based privacy preserving techniques can be grouped as randomized perturbation technique and randomized response technique for numeric and binary ratings respectively 2. Due to varying privacy concerns,users might perturb their confidential data variable or they might decide to use invariable data masking for perturbation
Johnson–Lindenstrauss Transformation method	1. This method preserves users' privacy without compromising utility.It guarantees user privacy by directly perturbing the original dataset using a transfer matrix 2. This method achieves ε-differential privacy with high accuracy
Modified Random Perturbation method	1. This is multi-level privacy preserving method by perturbing each rating before it is submitted to the server.The perturbation method is based on multiple levels and different ranges of random values for each level 2. Before the submission of each rating, the privacy level and the perturbation range are selected randomly from a fixed range of privacy levels 3. MAE = 1.3 for k = 20 (parameter)

each method can be configured accordingly, and each has its own unique characteristics. The proposed method can be used in various online environments based on user ratings to preserve user privacy by perturbing each rating before submission and providing acceptable accuracy recommendations. The proposed method's main achievement is that based on a randomly selected perturbation level for each rating, any potential attacker can confuse since it becomes more difficult to guess the perturbation range for a specific rating. Additionally, our approaches can preserve privacy, while the accuracy of the generated recommendations is acceptable. When applying the proposed method, the main implication is the potentially negative impact found on the accuracy when generating recommendations. When applying a perturbation method, the ratings are altered and usually differ from the real ones, leading to inaccurate computations of user similarity, leading to different nearest neighbors and inaccurate rating predictions, and eventually generated recommendations. Different

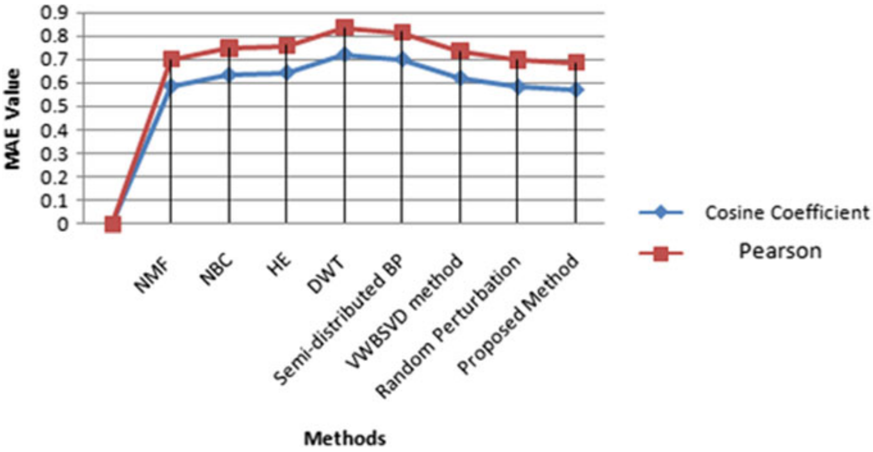

Fig. 4 Comparison of MAE among different methods using Cosine Coefficient and Pearson Correlation

perturbation methods might give different results, thus, protecting privacy at a different level. In our future work, we aim to investigate the employability of collaborative filtering methods in other systems to provide recommendations using deep learning.

References

1. Isinkaye, F.O., Folajimi, Y.O., Ojokoh, B.A.: Recommendation systems: principles, methods and evaluation. Egypt. Inf. J. **16**(3), 261–273 (2015)
2. Bobadilla, J., Ortega, F., Hernando, A., Gutiérrez, A.: Recommender systems survey. Knowl.-Based Syst. **46**, 109–132 (2013)
3. Ponnam, L.T., Punyasamudram, S.D., Nallagulla, S.N., Yellamati, S.: Movie recommender system using item based collaborative filtering technique. In: Proceedings of International Conference on Emerging Trends in Engineering, Technology and Science, pp. 1–5 (2016)
4. Burke, R., Felfernig, A., Göker, M.H.: Recommender systems: an overview. AI Mag. **32**(3), 13–18 (2011)
5. Ortega, F., Hernando, A., Bobadilla, J., Kang, J.H.: Recommending items to group of users using matrix factorization based collaborative filtering. Inf. Sci. **345**, 313–324 (2016)
6. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International Conference on World Wide Web, pp. 285–295, (2001)
7. Riyaz, P.A., Varghese, S.M.: A scalable product recommendations using collaborative filtering in hadoop for bigdata. Procedia Technol. **24**, 1393–1399 (2016)
8. Jooa, J., Bangb, S., Parka, G.: Implementation of a recommendation system using association rules and collaborative filtering. Procedia Comput. Sci. **91**, 944–952 (2016)
9. Ma, X., Lu, H., Gan, Z., Zeng, J.: An explicit trust and distrust clustering based collaborative filtering recommendation approach. Electron. Commer. Res. Appl. **25**, 29–39 (2017)
10. Polatidis, N., Georgiadis, C.K., Pimenidis, E., Mouratidis, H.: Privacy-preserving collaborative recommendations based on random perturbations. Expert Syst. Appl. **71**, 18–25 (2017)

11. Li, T., Gao, C., Du, J.: A NMF-based privacy-preserving recommendation algorithm. In: Proceedings of First IEEE International Conference on Information Science and Engineering, pp. 754–757 (2009)
12. Bilge, A., Polat, H.: Improving privacy-preserving NBC-based recommendations by preprocessing. In: Proceedings of IEEE International Conference on Web Intelligence and Intelligent Agent Technology, vol. 1, pp. 143–147 (2010)
13. Tada, M., Kikuchi, H., Puntheeranurak, S.: Privacy-preserving collaborative filtering protocol based on similarity between items. In: Proceedings of 24th IEEE International Conference on Advanced Information Networking and Applications, pp. 573–578 (2010)
14. Bilge, A., Polat, H.: An improved privacy-preserving DWT-based collaborative filtering scheme. Expert Syst. Appl. **39**(3), 3841–3854 (2012)
15. Luo, Z., Chen, S., Li, Y.: A distributed anonymization scheme for privacy-preserving recommendation systems. In: Proceedings of IEEE 4th International Conference on Software Engineering and Service Science, pp. 491–494 (2013)
16. Zou, J., Fekri, F.: A belief propagation approach to privacy-preserving item-based collaborative filtering. IEEE J. Sel. Topics Signal Proc. **9**(7), 1306–1318 (2015)
17. Batmaz, Z., Polat, H.: Randomization-based privacy-preserving frameworks for collaborative filtering. Procedia Comp. Sci. **96**, 33–42 (2016) Privacy-preserving Frameworks for Collaborative Filtering. Procedia Computer Science, **96**, 33–42 (2016)
18. Yang, M., Zhu, T., Ma, L., Xiang, Y., Zhou, W.: Privacy preserving collaborative filtering via the johnson-lindenstrauss transform. In: Proceedings of IEEE Trust, Security and Privacy in Computing and Communications, pp. 417–424 (2017)