J Intell Inf Syst (2016) 46:349–367
DOI 10.1007/s10844-015-0375-2

Robust recommendation method based on suspicious users measurement and multidimensional trust

Huawei Yi[1,2] · **Fuzhi Zhang**[1,2]

Received: 24 March 2015 / Revised: 18 July 2015 / Accepted: 20 July 2015 /
Published online: 14 August 2015

Abstract The existing collaborative recommendation algorithms have poor robustness against shilling attacks. To address this problem, in this paper we propose a robust recommendation method based on suspicious users measurement and multidimensional trust. Firstly, we establish the relevance vector machine classifier according to the user profile features to identify and measure the suspicious users in the user rating database. Secondly, we mine the implicit trust relation among users based on the user-item rating data, and construct a reliable multidimensional trust model by integrating the user suspicion information. Finally, we combine the reliable multidimensional trust model, the neighbor model and matrix factorization model to devise a robust recommendation algorithm. The experimental results on the MovieLens dataset show that the proposed method outperforms the existing methods in terms of both recommendation accuracy and robustness.

Keywords Shilling attacks · Robust recommendation · Multidimensional trust · Matrix factorization · Relevance vector machine · Analytic hierarchy process

This work was supported by the National Natural Science Foundation of China (No.61379116), the Natural Science Foundation of Hebei Province, China (No. F2013203124, No. F2015203046) , the Key Program of Research on Science and Technology of Higher Education Institutions of Hebei Province, China (No. ZH2012028) and the Scientific Research Foundation of Liaoning Provincial Education Department, China (No. L2015240).

✉ Fuzhi Zhang
xjzfz@ysu.edu.cn

[1] School of Information Science and Engineering, Yanshan University, Qinhuangdao 066004, China

[2] The Key Laboratory for Computer Virtual Technology and System Integration of Hebei Province, Qinhuangdao 066004, China

1 Introduction

As an important information filtering means, the collaborative recommender systems provide an effective way to solve the problem of “information overload” on the internet, and have been widely used in e-commerce websites (Xu et al. 2009). However, due to their natural openness, malicious users motivated by the commercial profit can inject a large number of fake profiles into the systems’ rating database in order to manipulate the recommendation results. Such behavior has been termed shilling attacks (Gunes et al. 2014), also called profile injection attacks (Aghili et al. 2011) or recommendation attacks (Zhang et al. 2006), and the fake user profiles injected are called attack profiles. Common shilling attack types include random attack, average attack, AoP attack and bandwagon attack (Burke et al. 2006). According to the purposes of attacks, shilling attacks can be divided into push attacks and nuke attacks (Hurley et al. 2007), which are used to increase and decrease the recommended frequency of target item respectively. A number of studies have indicated that collaborative recommender systems are vulnerable to shilling attacks, and the quality of recommendation will be harmed.

To reduce the influence of shilling attacks on the recommender systems, one way is to perform shilling attack detection before making recommendations. However, the existing shilling attack detection methods are based on binary classification, which is prone to filtering out the real user profiles, resulting in the decline of recommendation accuracy. An alternative way is to construct robust recommendation algorithms, i.e., to enhance the anti-attack ability of algorithms. In the field of recommender systems, robustness refers to the ability of a recommender system to provide stable recommendations when its rating database is contaminated with some portion of noisy or attack profiles (O’Mahony et al. 2004a). In this paper, we focus on developing a robust recommendation algorithm which not only has better anti-attack ability but also has higher recommendation accuracy.

In recent years, research on the robustness of recommendation algorithms has been conducted in the context of shilling attacks. A neighbor selection method is proposed by O’Mahony (2004). First, the active user’s neighbors are divided into two groups by clustering according to the rating of the target item. Then, the standard deviations of the target item are calculated respectively, and the group of users with smaller standard deviation is deemed as the attack users. O’Mahony et al. (2004b) propose a profile utility calculation method based on the item’s inverse popularity to adjust the similarity weight between users, and then an intelligent neighborhood formation method is proposed to reduce the influence of attacks on recommendation results. The trust is introduced into collaborative recommendation, and a trust measurement method is proposed by Weng et al. (2006). Compared with the conventional similarity-based recommendation algorithms, the trust-based algorithm can improve the recommendation accuracy, coverage and robustness. O’Donovan and Smyth (2005) propose the profile-level and item-level computational models of trust, and the experimental results show that the latter performs better than the former in recommendation accuracy. Pitsilis and Marshall (2004) propose a trust computation method based on the uncertain probability theory. This method analyzes the trust relation between users from the perspective of subjective logic thinking. A multidimensional credibility model based on the source credibility theory is proposed by Kwon et al. (2009). However, it only takes into account the heterogeneous of ratings of users and still has the vulnerability when there are shilling attacks in recommender systems. Based on the theoretical analysis of knowledge-based trustworthiness and deduction trustworthiness, a multidimensional trustworthiness model is proposed by Maida et al. (2012), but it does not give the specific calculation method.

Mobasher et al. (2006) propose two recommendation algorithms.One is based on k-means clustering and the other is based on probabilistic latent semantic analysis (PLSA). Compared with standard k-neighbor approach, they have better robustness. Furthermore, the PLSA-based algorithm can achieve comparable recommendation accuracy. The recommendation algorithm based on association rule mining is presented by Sandvig et al. (2007). This algorithm can get better robustness, but the robustness is acquired at the cost of coverage. Mehta et al. (2007) propose a matrix factorization algorithm based on M-estimator (MMF), which can resist the outliers to some extent in comparison with PLSA and k-neighbor. But this method only works on moderate attacks. The least trimmed squares estimator based matrix factorization (LTSMF) (Cheng and Hurley 2010) shows better robustness and accuracy compared with MMF. LTS-estimator trims part of the largest residuals, which may cause the loss of recommendation accuracy.

In this paper, we propose a robust recommendation method based on suspicious users measurement and multidimensional trust (RRM-SUMMT). The main contributions are summarized as follows:

- According to the user profiles features, we construct the RVM-based classifier to identify and measure the suspicious users in the user rating database and get the user suspicion degree.
- Based on the user-item rating data of the recommendation system, we mine the implicit trust relationships between users and combine with the user suspicion degree to construct a reliable multidimensional trust model.
- We design a robust recommendation algorithm which incorporates reliable multidimensional trust model, neighbor model and matrix factorization model and demonstrate the effectiveness of the proposed algorithm.

2 Background

In this section, we first introduce the theory of relevance vector machine. Then, we present the theory of matrix factorization.

2.1 Relevance vector machine

Relevance vector machine (RVM) is a machine learning technique based on Bayesian theory framework, which is proposed by Tipping in 2000. Similar to the support vector machine (SVM), RVM also converts the nonlinear problem of low-dimensional space into the linear problem of high-dimensional space based on the kernel function mapping (Tipping 2001a, b). Compared with SVM, RVM has the following advantages: a) the selection of kernel function is not limited by Mercer condition; b) the parameters can be obtained without cross verification; c) it has good generalization ability; d) there are fewer relevance vectors and the model is sparser; e) it provides the probability of prediction which can be used to analyze the uncertainty of the problem.

For binary classification, a training set $\{(x_i, t_i), i = 1, 2, ..., N\}$ is given, in which N is the number of sample, and the corresponding objective output is $t_i \in \{0, 1\}$, where 0 and 1 represent the category label of the two kinds of samples, respectively. The prediction model of RVM maps the linear combination $y(x)$ to the interval (0, 1) by means of Sigmoid function for the judgment of category:

$$\sigma(y) = 1/(1 + e^{-y}) \tag{1}$$

where $y(x)$ is the output of RVM, which is the linear combination of weight vector and nonlinear kernel function:

$$y(\boldsymbol{x}) = \sum_{i=1}^{N} w_i k(\boldsymbol{x}, \boldsymbol{x}_i) \tag{2}$$

where $k(\boldsymbol{x}, \boldsymbol{x}_i)$ is a kernel function, and w_i represents the weight.

Assume that the samples are independent identically distributed, and $p(\boldsymbol{t}|\cdot)$ obeys Bernoulli distribution, and then the likelihood function of the sample set can be expressed as:

$$p(\boldsymbol{t}|\boldsymbol{w}) = \prod_{i=1}^{N} \sigma\{y(\boldsymbol{x}_i; \boldsymbol{w})\}^{t_i} [1 - \sigma\{y(\boldsymbol{x}_i; \boldsymbol{w})\}]^{1-t_i} \tag{3}$$

If the maximum likelihood estimation method is employed in the formula (3), it is likely to generate the problem of over-fitting. To avoid this, RVM assumes that the parameter w_i obeys the Gaussian conditional probability distribution with mean value of 0 and variance α_i^{-1}, therefore:

$$p(\boldsymbol{w}|\boldsymbol{\alpha}) = \prod_{i=1}^{N} N(w_i|0, \alpha_i^{-1}) \tag{4}$$

where, $\boldsymbol{\alpha}$ is the hyper-parameter vector deciding the prior distribution of the weight vector $\boldsymbol{w}$, which controls the degree of deviation from zero mean of each weight value.

According to the Bayesian theory, if the likelihood distribution of sample set and the prior probability distribution of the weight value are known, then the posterior probability of the model parameter is:

$$p(\boldsymbol{w}, \boldsymbol{\alpha}|\boldsymbol{t}) = p(\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\alpha}) p(\boldsymbol{\alpha}|\boldsymbol{t}) \tag{5}$$

For the sample x_* to be detected, the prediction distribution of the corresponding output t_* is:

$$p(t_*|\boldsymbol{t}) = \int p(t_*|\boldsymbol{w}, \boldsymbol{\alpha}) p(\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\alpha}) p(\boldsymbol{\alpha}|\boldsymbol{t}) d\boldsymbol{w} d\boldsymbol{\alpha} \tag{6}$$

In the above formula (6), the weight posterior probability $p(\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\alpha})$ and marginal likelihood function $p(\boldsymbol{\alpha}|\boldsymbol{t})$ can not be directly solved through integral, so we use the Laplace method proposed by Mackay (1992) for approximation.

2.2 Matrix factorization

Matrix factorization model can reveal the hidden characteristics of users and items in ratings data, which is denoted by user feature matrix $\boldsymbol{P}$ and item feature matrix $\boldsymbol{Q}$. Let $\hat{\boldsymbol{R}}$ be the matrix of predicted ratings, and then the matrix factorization model is defined as follows:

$$\hat{\boldsymbol{R}} = \boldsymbol{Q}^T \boldsymbol{P} \tag{7}$$

where $\boldsymbol{P} = (\boldsymbol{p_1}, \boldsymbol{p_2}, ..., \boldsymbol{p_m})$ is the $f \times m\,(f < m)$ matrix, and $\boldsymbol{p_u}$ is the f-dimensional feature vector for user u, $\boldsymbol{Q} = (\boldsymbol{q_1}, \boldsymbol{q_2}, ..., \boldsymbol{q_n})$ is the $f \times n\,(f < n)$ matrix, and $\boldsymbol{q_i}$ is the f-dimensional feature vector for item i. Let $\hat{r}_{u,i}$ be the predicted rating, which is expressed as follows:

$$\hat{r}_{u,i} = \boldsymbol{q_i}^T \boldsymbol{p_u} \tag{8}$$

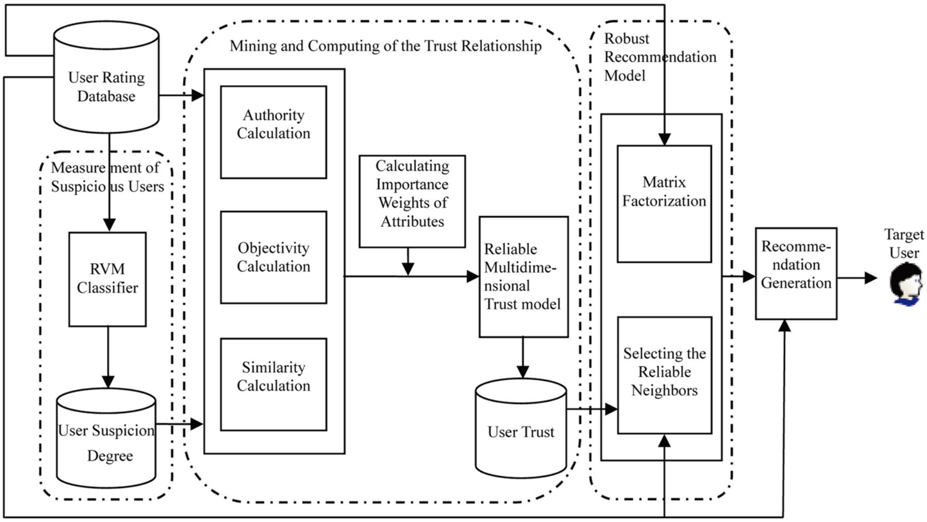
User Rating Database
Mining and Computing of the Trust Relationship
Authority Calculation
Objectivity Calculation
Similarity Calculation
Calculating Importance Weights of Attributes
Reliable Multidimensional Trust model
User Trust
Measurement of Suspicious Users
RVM Classifier
User Suspicion Degree
Robust Recommendation Model
Matrix Factorization
Selecting the Reliable Neighbors
Recommendation Generation
Target User

Fig. 1 Framework of the robust recommendation algorithm RRM-SUMMT

In order to solve the vectors $\boldsymbol{p_u}$ and $\boldsymbol{q_i}$, the least squares problem is defined as follows:

$$\boldsymbol{P^*}, \boldsymbol{Q^*} = \arg\min \sum_{r_{u,i} \neq \phi} (r_{u,i} - \boldsymbol{q_i^T} \boldsymbol{p_u})^2 + \lambda(\|\boldsymbol{q}_i\|^2 + \|\boldsymbol{p}_u\|^2) \tag{9}$$

where $r_{u,i}$ is the real rating, $\lambda(\|\boldsymbol{q_i}\|^2 + \|\boldsymbol{p_u}\|^2)$ is a regularization term which can avoid overfitting and λ is a constant.

We use stochastic gradient descent to solve the optimization problem above, and its steps are defined as follows:

$$\boldsymbol{q_i} \leftarrow \boldsymbol{q_i} + \gamma(\boldsymbol{p_u} e_{u,i} - \lambda \boldsymbol{q_i}) \tag{10}$$

$$\boldsymbol{p_u} \leftarrow \boldsymbol{p_u} + \gamma(\boldsymbol{q_i} e_{u,i} - \lambda \boldsymbol{p_u}) \tag{11}$$

where $e_{u,i}$ is the residual between real rating and predicted rating in recommender systems, which is computed as follows:

$$e_{u,i} = r_{u,i} - \hat{r}_{u,i} \tag{12}$$

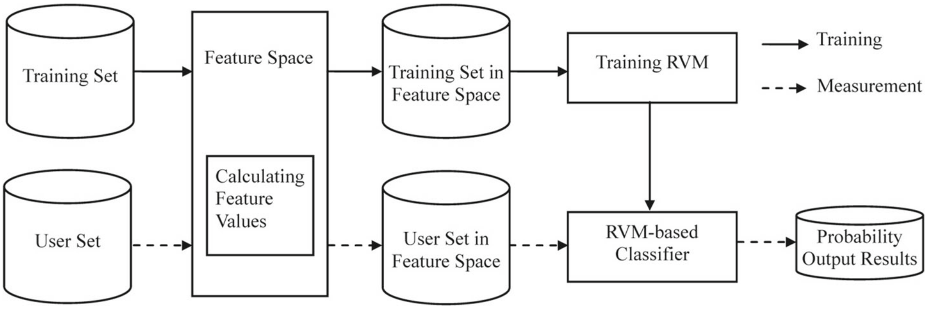
Training Set
User Set
Feature Space
Calculating Feature Values
Training Set in Feature Space
User Set in Feature Space
Training RVM
RVM-based Classifier
Probability Output Results
Training
Measurement

Fig. 2 Framework of the suspicious users measurement based on RVM

3 The proposed method

To ensure the quality of recommendation, we first use RVM-based classifier to identify and measure the suspicious users in rating database and get the user suspicion degree. Then we mine the implicit trust relationships between users and incorporate the user suspicion degree to construct a reliable multidimensional trust model. Finally, we combine the trust model, neighbor model and matrix factorization to design a robust recommendation algorithm RRM-SUMMT. The framework of RRM-SUMMT is shown in Fig. 1.

3.1 Suspicious users measurement based on RVM

In this Section, we propose a RVM-based method for the suspicious users measurement (SUM_RVM). As shown in Fig. 2, SUM_RVM consists of two stages: training stage of RVM-based classifier and measurement stage of suspicious users.

In the process of training, the training set includes genuine and attack profiles which are marked with 0 and 1, respectively. In order to generate training set and user profile set to be measured in feature space, we use the following features proposed by Williams (2007a, b).

- 6 generic features: WDMA, RDMA, WDA, Length Variance, DegSim and DegSim';
- 3 Average attack model features(3 for push): FMV, FMD, PV;
- 2 Random attack model features(2 for push): FAC, FMD;
- 2 Bandwagon attack model features(2 for push): FAC, FMD;

The training set is expressed as the form of feature vectors. We can employ the training set composed by the feature vectors to train the relevance vector machine and generate RVM-based classifier.

In the process of measurement, the user profile set to be measured is expressed as feature vectors set. So, we can employ RVM-based classifier to get the prediction probability of user classification (we call it user suspicion degree). The larger prediction probability is, the larger the user suspicion degree is.

Let $Susdegree_{set}$ denote the set of measurement results. The algorithm of suspicious users measurement based on RVM (SUM_ RVM) is described as follows.

Algorithm 1 SUM_ RVM

```
Input: training set $Train_{set}$
       user profile set to be measured $User_{set}$
Output: $Susdegree_{set}$
Begin
1 Generate training set $Set_{Train}$ and user profile set to be measured $Set_{User}$ in feature space
  by use of the 13 profile features mentioned above to extract features of each user $u$ in
  $Train_{set}$ and $User_{set}$;
2 Generate the $RVM\_Classifier$ by using $set_{Train}$ to train RVM;
3 Generate the suspicious degree of every user in $Set_{User}$ by using $RVM\_Classifier$, and
  the suspicious degrees are stored in $Susdegree_{set}$;
4 return $Susdegree_{set}$;
End
```

3.2 Reliable multidimensional trust model

In this section, we introduce three trust attributes (i.e., rating authority, rating objectivity, and rating similarity) and combine the user suspicion degree with them to construct a reliable multidimensional trust model.

Definition 1 (Rating authority) The rating authority of $v \in U$, denoted as $A(v)$, is defined as follows:

$$A(v) = \frac{\sum_{i \in I_v} Trust\_item(v, i)}{|I_v|} \times (1 - sus\_degree(v)) \tag{13}$$

$$Trust\_item(v, i) = \frac{\sum_{c \in U_i} Correct_v(c, i)}{|U_i|} \tag{14}$$

$$Correct_v(c, i) = \begin{cases} 1 \, , & |p_{c,i} - r_{c,i}| < \varepsilon \\ 0 \, , & others \end{cases} \tag{15}$$

where $p_{c,i}$ represents the prediction rating of user c on item i under the condition of the user v is as the sole recommendation user of user c ; $r_{c,i}$ represents the real rating of user c on item i ; I_v is the set of items rated by user v ; for item $i \in I_v$, the set of users who rate on item i is expressed as $U_i = \{c | c \neq v, c \in U, r_{c,i} \neq 0\}$;$sus\_degree(v)$ represents the suspicion degree of user v; $\varepsilon = 1.8$ (O'Donovan and Smyth 2005).

Definition 2 (Rating objectivity) The rating objectivity of user $v \in U$, denoted as $O(v)$, is defined as follows:

$$O(v) = \left(1 - \frac{\sum_{i \in I_v} |r_{v,i} - \bar{r}_i|}{|I_v|}\right) \times (1 - sus\_degree(v)) \tag{16}$$

where, $\bar{r}_i$ is the average rating of item i ; I_v is the set of items rated by user v ; $sus\_degree(v)$ represents the suspicion degree of user v.

Definition 3 (Rating similarity.) For user $u \in U$ and $v \in U$, the rating similarity between user u and user v, denoted as $S(u, v)$, is defined as follows:

$$S(u, v) = \frac{\sum_{i \in I(u,v)} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I(u,v)} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I(u,v)} (r_{v,i} - \bar{r}_v)^2}} \times (1 - sus\_degree(v)) \tag{17}$$

Where, $\bar{r}_u$ and $\bar{r}_v$ are the average ratings of user u and user v respectively; $I(u, v)$ is the set of items co-rated by user u and user v; $sus\_degree(v)$ represents the suspicion degree of user v.

Based on the above definitions, we propose the computing method of reliable multidimensional trust model, which is regarded as the sum of the trust attribute and its importance weight, the trust value of user u for user v is shown in the following equation:

$$Trust_{u,v} = w_A A(v) + w_O O(v) + w_S S(u, v) \tag{18}$$

where, w_A, w_O and w_S represent the importance weight of rating authority, rating objectivity and rating similarity, respectively. We use analytic hierarchy process to compute the importance weight of each trust attribute.

Analytic hierarchy process (AHP) (Dyer 1992) is a hierarchy weight decision-making analysis method proposed by American operational research expert T. L. Saaty. AHP can mathematize the thinking process of decision-making with less quantitative information, thus providing a simple decision-making method for the multi-objective and multi-criteria complicated decision-making problem.

The main steps are as follows:

Step 1: We implement three recommendation strategies based on the above trust attributes (i.e., rating authority, rating objectivity, rating similarity). For each of the recommendation strategies, we calculate its recommendation accuracy, coverage and prediction shift, respectively.

Step 2: Take the recommendation accuracy, coverage and prediction shift as the assessment criteria of the importance of trust attributes and construct the hierarchy model.

Step 3: Compare the recommendation accuracy, coverage and prediction shift of three recommendation strategies respectively and use “1-9 scale method” to construct the pairwise comparison matrices.

Step 4: If the pairwise comparison matrices satisfy the consistency verification, we compute the weight vectors of pairwise comparison matrices using geometric mean method. Then we get the weight vectors of the trust attributes aiming at the above assessment criteria.

Step 5: Aiming at the trust attributes, we compute the combination of weight vectors and get the importance weight $w = [w_1\ w_2\ w_3]^T$, where w_1, w_2 and w_3 denote the importance weight of rating authority, rating objectivity and rating similarity, respectively.

Based on the above multidimensional trust model, we can calculate the degree of trust between users. The algorithm of user trust computation (UTC) is described as follows:

Algorithm 2 UTC

```
Input: user-item rating matrix $\boldsymbol{R}$
       the set of suspicion degree $Susdegree_{set}$
Output: the trust $Trust_{u,v}$ of user $u$ for user $v$
Begin
1 Get the suspicion degree of user $v$ from $Susdegree_{set}$;
2 Compute the rating authority $A(v)$ of user $v$ using formula (13);
3 Compute the rating objectivity $O(v)$ of user $v$ using formula (16);
4 Compute the rating similarity $S(u, v)$ between user $u$ and user $v$ using formula (17);
5 $Trust_{u,v} \leftarrow w_A A(v) + w_O O(v) + w_S S(u, v)$;
6 return $Trust_{u,v}$ ;
End
```

Algorithm 2 mainly includes two parts: the first part, from lines 1 to 4, is to get the suspicion degree of user v and compute user v’ s rating authority, rating objectivity and rating similarity. The second part, from lines 5 to 6, is to compute the trust of user u to user v and return the trust value.

3.3 Robust recommendation algorithm

By combining the multidimensional trust model, matrix factorization model and neighbor model, we propose a robust recommendation algorithm based on suspicious users measurement and multidimensional trust (RRM-SUMMT). The formula of prediction rating can be written as

$$\hat{r}_{u,i} = \mu + b_u + b_i + \boldsymbol{q_i^T} \boldsymbol{p_u} + |R(u)|^{-\frac{1}{2}} \sum_{v \in R(u)} (r_{v,i} - \bar{r}_v) Trust_{u,v} \tag{19}$$

where, μ refers to the average value of all ratings in rating database; b_u and b_i indicate the observed deviations of user u and item i, respectively, from the average; $R(u)$ is the set of similar users for the target user u, $r_{v,i}$ is the rating of user v to item i, $\bar{r}_v$ is the average rating of user v, $Trust_{u,v}$ is the trust of user u to user v.

$$b_u = \frac{1}{|I_u|} \sum_{i \in I_u} (r_{u,i} - \mu) \tag{20}$$

$$b_i = \frac{1}{|U_i|} \sum_{u \in U_i} (r_{u,i} - b_u - \mu) \tag{21}$$

where, I_u is the set of items rated by the user u; U_i is the set of users who rate on item i.

Formula (19) can be solved by stochastic gradient descent, and the iterative formulae are as follows:

$$\boldsymbol{q_i} \leftarrow \boldsymbol{q_i} + \gamma (e_{u,i} \boldsymbol{p_u} - \lambda \boldsymbol{q_i}) \tag{22}$$

$$\boldsymbol{p_u} \leftarrow \boldsymbol{p_u} + \gamma (e_{u,i} \boldsymbol{q_i} - \lambda \boldsymbol{p_u}) \tag{23}$$

$$b_u \leftarrow b_u + \gamma (e_{u,i} - \lambda b_u) \tag{24}$$

$$b_i \leftarrow b_i + \gamma (e_{u,i} - \lambda b_i) \tag{25}$$

The core idea of RRM-SUMMT is summarized as follows:

Algorithm 3 RRM-SUMMT

```
Input: user-item rating matrix $\boldsymbol{R}$
Output: the predicted rating $\hat{r}_{t,j}$ for target user $t$ on target item $j$
Begin
 1 Initialize feature matrix $\boldsymbol{P}$, $\boldsymbol{Q}$ ;
 2 for each item $i \in I$ do
 3   for each user $u \in U$ do
 4     compute $b_u$ ;
 5     compute $b_i$ ;
 6   end for
 7 end for
 8 repeat
 9   for each user $u \in U$ do
10     for each item $i \in I$ do
11       if $r_{u,i} \neq 0$ then
12         $Trust_{u,v} \leftarrow UTC(u, v)$;
13         $\hat{r}_{u,i} \leftarrow b_{u,i} + \boldsymbol{q_i}^T \boldsymbol{p_u} + |R(u)|^{-\frac{1}{2}} \sum_{v \in R(u)} (r_{v,i} - \bar{r}_v) Trust_{u,v}$;
14         $e_{u,i} \leftarrow r_{u,i} - \hat{r}_{u,i}$;
15         for $k$=1 to $f$ do
16           $\boldsymbol{q}_{i,k} \leftarrow \boldsymbol{q}_{i,k} + \gamma(e_{u,i}\boldsymbol{p}_{u,k} - \lambda\boldsymbol{q}_{i,k})$;
17           $\boldsymbol{p}_{u,k} \leftarrow \boldsymbol{p}_{u,k} + \gamma(e_{u,i}\boldsymbol{q}_{i,k} - \lambda\boldsymbol{p}_{u,k})$;
18         end for
19         $b_u \leftarrow b_u + \gamma(e_{u,i} - \lambda b_u)$ ;
20         $b_i \leftarrow b_i + \gamma(e_{u,i} - \lambda b_i)$ ;
21       end if
22     end for
23   end for
24 until $\boldsymbol{P}$, $\boldsymbol{Q}$ no longer changes
25 compute the predicted rating $\hat{r}_{t,j}$ using Formula (19);
26 return $\hat{r}_{t,j}$ ;
End
```

Algorithm 3 mainly includes three parts: the first part, from lines 1 to 7, is to initialize user feature matrix and item feature matrix and calculate the b_u and b_i. The second part, from lines 8 to 24, is to complete model training process and obtain the optimal user feature matrix and item feature matrix. The third part, from lines 25 to 26, is to compute the predicted rating $\hat{r}_{t,j}$ for user t on item j.

4 Experimental evaluations

4.1 Dataset

We use the MovieLens 100K dataset as the experimental data Miller et al. (2003). This dataset consists of 100,000 ratings on 1682 movies by 943 users. All ratings are integer values between 1 and 5, where 1 is the lowest (disliked) and 5 is the highest (most liked).

Table 1 Comparison of MAE for five algorithms with random attack

filler size	3 %						5 %					
attack size	1 %	2 %	4 %	6 %	8 %	10 %	1 %	2 %	4 %	6 %	8 %	10 %
MMF	0.7485	0.7492	0.7491	0.7503	0.7508	0.7516	0.7491	0.7483	0.7486	0.7484	0.7486	0.7497
LTSMF	0.7465	0.7461	0.7472	0.7461	0.7469	0.7457	0.7467	0.7471	0.7479	0.7477	0.7487	0.7492
CF	0.7587	0.7589	0.7594	0.7608	0.7611	0.7626	0.7582	0.7597	0.7600	0.7612	0.7626	0.7633
WItem	0.7555	0.7551	0.7557	0.7552	0.7554	0.7561	0.7553	0.7552	0.7554	0.7569	0.7566	0.7585
RRM-SUMMT	0.7413	0.7401	0.7398	0.7391	0.7377	0.7378	0.7422	0.7395	0.7384	0.7390	0.7378	0.7379

Table 2 Comparison of MAE for five algorithms with average attack

filler size	3 %						5 %					
attack size	1 %	2 %	4 %	6 %	8 %	10 %	1 %	2 %	4 %	6 %	8 %	10 %
MMF	0.7493	0.7488	0.7494	0.7510	0.7507	0.7495	0.7497	0.7497	0.7505	0.7509	0.7489	0.7494
LTSMF	0.7464	0.7463	0.7459	0.7471	0.7481	0.7469	0.7470	0.7474	0.7453	0.7475	0.7488	0.7569
CF	0.7585	0.7595	0.7608	0.7608	0.7616	0.7617	0.7587	0.7593	0.7591	0.7623	0.7621	0.7620
WItem	0.7554	0.7554	0.7562	0.7565	0.7570	0.7572	0.7550	0.7563	0.7567	0.7582	0.7585	0.7585
RRM-SUMMT	0.7421	0.7419	0.7420	0.7409	0.7412	0.7405	0.7416	0.7411	0.7413	0.7417	0.7416	0.7401

Table 3 Comparison of MAE for five algorithms with bandwagon attack

filler size	3 %						5 %					
attack size	1 %	2 %	4 %	6 %	8 %	10 %	1 %	2 %	4 %	6 %	8 %	10 %
MMF	0.7495	0.7491	0.7493	0.7501	0.7503	0.7491	0.7480	0.7496	0.7509	0.7500	0.7526	0.7519
LTSMF	0.7482	0.7461	0.7466	0.7470	0.7461	0.7476	0.7468	0.7466	0.7479	0.7472	0.7490	0.7503
CF	0.7587	0.7587	0.7596	0.7603	0.7616	0.7627	0.7585	0.7597	0.7600	0.7606	0.7632	0.7623
WItem	0.7552	0.7553	0.7547	0.7567	0.7568	0.7563	0.7551	0.7557	0.7561	0.7562	0.7581	0.7571
RRM-SUMMT	0.7416	0.7416	0.7401	0.7398	0.7396	0.7386	0.7411	0.7405	0.7395	0.7391	0.7397	0.7388

Table 4 Comparison of MAE for five algorithms with AoP attack

filler size	3 %						5 %					
attack size	1 %	2 %	4 %	6 %	8 %	10 %	1 %	2 %	4 %	6 %	8 %	10 %
MMF	0.7494	0.7479	0.7479	0.7472	0.7484	0.7494	0.7483	0.7500	0.7489	0.7491	0.7491	0.7487
LTSMF	0.7455	0.7477	0.7455	0.7469	0.7477	0.7461	0.7451	0.7479	0.7471	0.7464	0.7471	0.7483
CF	0.7583	0.7583	0.7587	0.7594	0.7589	0.7596	0.7581	0.7586	0.7587	0.7589	0.7592	0.7597
WItem	0.7552	0.7553	0.7555	0.7558	0.7556	0.7560	0.7552	0.7554	0.7554	0.7557	0.7561	0.7564
RRM-SUMMT	0.7419	0.7416	0.7430	0.7422	0.7409	0.7412	0.7416	0.7423	0.7421	0.7411	0.7417	0.7420

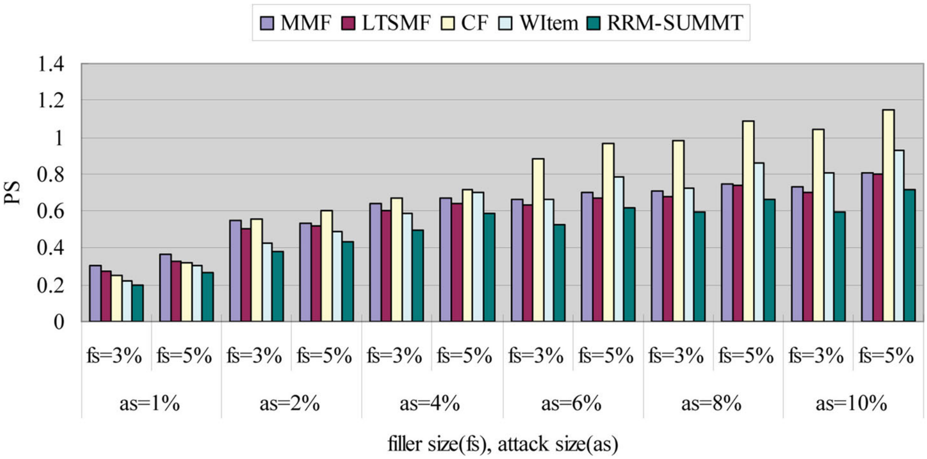

Fig. 3 Comparison of PS for five algorithms with random attack

For the number of items which one user has rated in the dataset, the minimum is 20. The dataset is divided randomly in a ratio 80:20 into training and test sets.

4.2 Evaluation metrics

We use mean absolute error (MAE) and prediction shift (PS) to evaluate the performance of RRM-SUMMT.

MAE is commonly used in recommender systems as the measurement of accuracy, and it is defined as follows (Kantor et al. 2011):

$$MAE = \frac{\sum_{j=1}^{n} \left| r_{u,i} - \hat{r}_{u,i} \right|}{N} \tag{26}$$

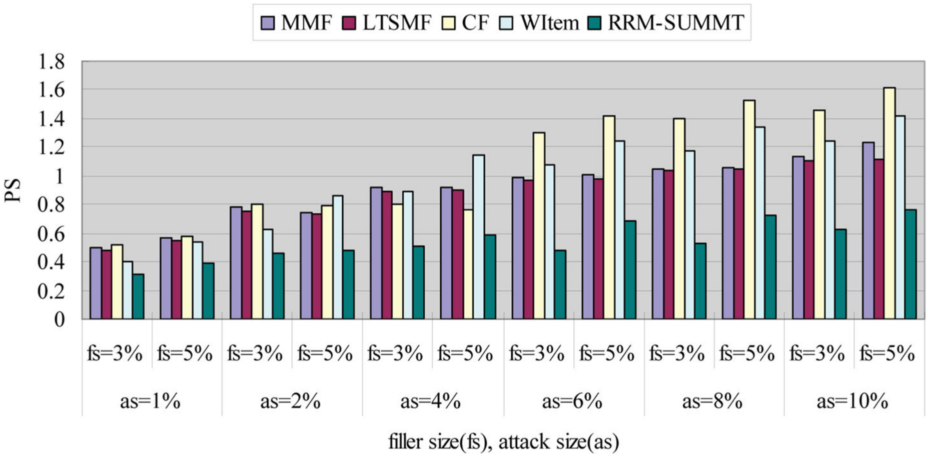

Fig. 4 Comparison of PS for five algorithms with average attack

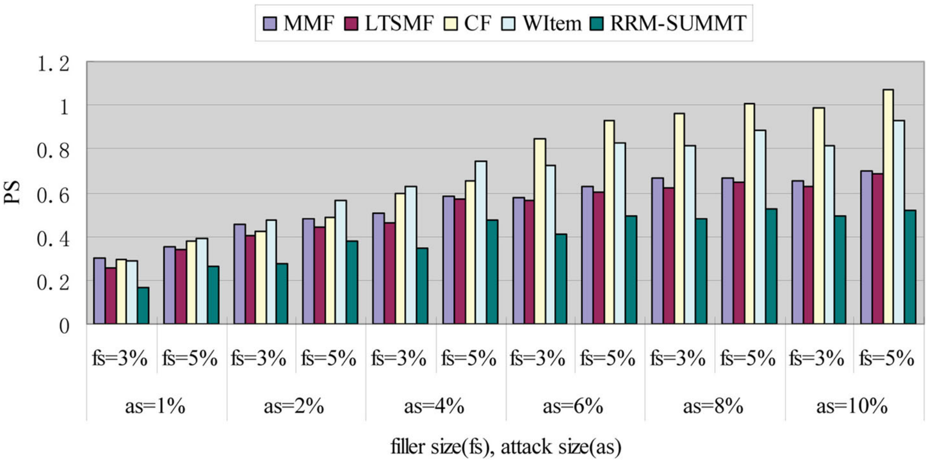

Fig. 5 Comparison of PS for five algorithms with bandwagon attack

where $r_{u,i}$ is the real rating of user u on item i, $\hat{r}_{u,i}$ is the predicted rating of user u on item i, N is the total number of prediction.

PS is commonly used in recommender systems as the measurement of robustness, and it is computed as follows (Kantor et al. 2011):

$$PS = \frac{\sum\limits_{u \in U} \left| \hat{r}'_{u,i} - \hat{r}_{u,i} \right|}{N} \tag{27}$$

where $\hat{r}_{u,i}$ and $\hat{r}'_{u,i}$ are the predicted ratings of user u on item i before and after the target item i is attacked respectively, N is the total number of prediction.

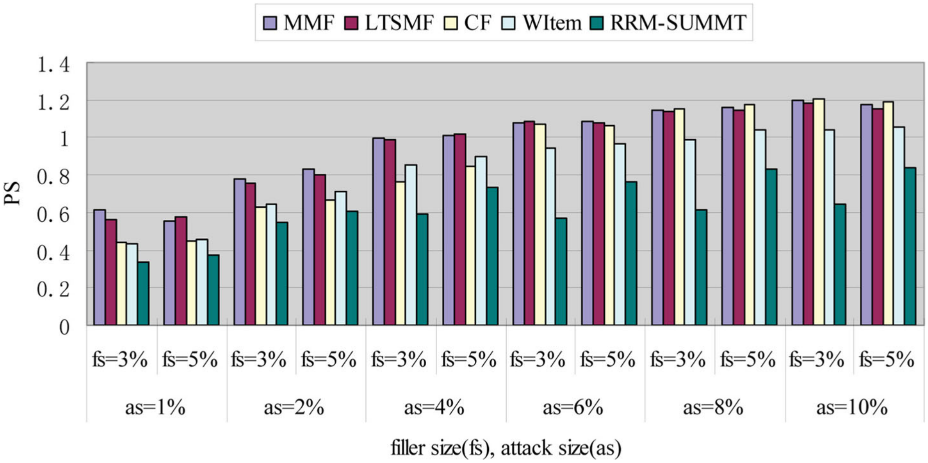

Fig. 6 Comparison of PS for five algorithms with AoP attack

4.3 Experimental results and analysis

To evaluate the performance of algorithm, we compare the performance of RRM-SUMMT with the following methods.

(1) MMF: Matrix factorization algorithm based on M-estimator proposed by Mehta et al. (2007).
(2) LTSMF: Matrix factorization algorithm based on least trimmed square estimator proposed by Cheng and Hurley (2010).
(3) CF: User-based collaborative filter recommendation algorithm.
(4) WItem: Recommendation algorithm based on trust proposed by O'Donovan and Smyth (2005).

To evaluate the robustness of algorithm, average attack, random attack, bandwagon attack and AoP attack are respectively injected into the training set. The attack size is 1 %, 2 %, 4 %, 6 %, 8 % and 10 %. The filler size is 3 % and 5 %. The results of experimental data comparison for the five algorithms are shown in Tables 1, 2, 3 and 4 and Figs. 3, 4, 5 and 6.

From Tables 1 to 4, we can see that under the same attack size and filler size, MAE value of RRM-SUMMT is best. Take MAE value under random attack with the filler size of 5 % for an example. Compared with MMF, LTSMF, CF and WItem, MAE values of RRM-SUMMT reduce 1.3 %, 1.2 %, 2.9 % and 2.3 %, respectively. On the whole, MAE values of MMF and LTSMF are between 0.7457 and 0.7516; MAE values of WItem and CF are between 0.7551 and 0.7633; MAE value of RRM-SUMMT is between 0.7377 and 0.7422. Since lower MAE value means better recommendation accuracy, the recommendation accuracy of RRM-SUMMT is superior to MMF, LTSMF, CF and WItem. It thus can be seen that the combination of the neighbor model with matrix factorization can improve the recommendation accuracy of algorithm.

From Figs. 3 to 6, it can be seen that under the four attack types mentioned in this paper, when the filler size is the same, the PS values of the five algorithms rise with the increase of attack size. Thus, the anti-attack ability of the algorithms gradually drops with the increase of the attack users. Under the condition of the same filler size and the same attack size, compared with MMF, LTSMF, CF and WItem, the PS value of RRM-SUMMT is obviously small. Take PS value of random attack with 3 % filler size for example. Compared with MMF, LTSMF, CF and WItem, PS values of RRM-SUMMT reduce 22.4 %, 17.4 %, 36.2 % and 18.3 %, respectively.

Under the four attack situations, the PS values of CF and WItem increase obviously; the PS values of MMF and LTSMF increase obviously along with the attack size increases under Aop attack. However, the PS values of RRM-SUMMT are not change obviously under the four attack situations. Lower PS value means better robustness. Thus, RRM-SUMMT has strong anti-attack ability, i.e. good robustness. The reason is that the RRM-SUMMT is incorporated with multidimensional trust model. Therefore, the neighbors reliability of target user is improved greatly so as to reduce the influence of attack profiles on the recommendation results. So, the robustness of RRM-SUMMT is improved.

5 Conclusions and future work

Improving the robustness of algorithm is an effective approach to increase the recommender systems' anti-attack ability. In this paper, we propose a robust recommendation algorithm based on the suspicious user measurement and multidimensional trust. First, in order to measure the possibility of user as attacker, we propose a RVM-based suspicious user measurement method. Then, we mine the user trust attributes from user rating authority, rating objectivity and rating similarity, and the suspicion degree is incorporated as the weight factor of attributes. We propose a reliable multidimensional trust model to measure the user's reliability. We combine the multidimensional trust model, matrix factorization model and neighbor model. The experiment demonstrates that the proposed algorithm does not only increase the robustness, but also improves the accuracy.

Future work can be carried out from the following two aspects. On the one hand, we will focus on improving the reliable multidimensional trust model proposed in this paper. In particular, we will propose more effective features to characterize the attack profiles, and use ensemble method to detect shilling attacks in order to further improve the performance of classification. On the other hand, we will introduce the items' attribute information to address the data sparsity issue in order to further improve the recommendation accuracy. Particularly, we will explore more effective method to complete the missing values of user rating matrix.

References

Aghili, G., Shajari, M., Khadivi, S., & Morid, M.A. (2011). Using genre interest of users to detect profile injection attacks in movie recommender systems. In *Proceedings of the 10th international conference on Machine Learning and Applications and Workshops* (pp. 49–52).

Burke, R., Mobasher, B., Williams, C., & Bhaumik, R. (2006). Classification features for attack detection in collaborative recommender systems. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 542–547).

Cheng, Z., & Hurley, N. (2010). Robust collaborative recommendation by least trimmed squares matrix factorization. In *Proceedings of the 22nd IEEE international conference on tools with artificial intelligence* (pp. 105–112).

Dyer, R.F. (1992). Group decision support with the analytic hierarchy process. *Decision Support Systems*, *8*(2), 99–124.

Gunes, I., Kaleli, C., Bilge, A., & Polat, H. (2014). Shilling attacks against recommender systems: a comprehensive survey. *Artificial Intelligence Review*, *42*(4), 767–799.

Hurley, N.J., O'Mahony, M.P., & Silvestre, G.C. (2007). Attacking recommender systems: A cost-benefit analysis. *IEEE Intelligent Systems*, *22*(3), 64–68.

Kantor, P.B., Rokach, L., Ricci, F., & Shapira, B. (2011). *Recommender systems handbook.* Springer.

Kwon, K., Cho, J., & Park, Y. (2009). Multidimensional credibility model for neighbor selection in collaborative recommendation. *Expert Systems with Applications*, *36*(3), 7114–7122.

Mackay, D. (1992). The evidence framework applied to classification networks. *Neural computation*, *4*(5), 720–736.

Maida, M., Maier, K., Obwegeser, N., & Stix, V. (2012). A multidimensional model of trust in recommender systems. In *Proceedings of 13th International Conference on Electronic Commerce and Web Technologies* (pp. 212–219).

Mehta, B., Hofmann, T., & Nejdl, W. (2007). Robust collaborative filtering. In *Proceedings of the 2007 ACM conference on Recommender systems* (pp. 49–56).

Miller, B.N., Albert, I., Lam, S.K., Konstan, J.A., & Riedl, J. (2003). MovieLens unplugged: experiences with an occasionally connected recommender system. In *Proceedings of the 8th international conference on Intelligent user interfaces* (pp. 263–266).

Mobasher, B., Burke, R., & Sandvig, J.J. (2006). Model-based collaborative filtering as a defense against profile injection attacks. In *Proceedings of the 21st national conference on artificial intelligence* (pp. 1388–1393).

O'Donovan, J., & Smyth, B. (2005). Trust in recommender systems. In *Proceedings of the 10th international conference on Intelligent user interfaces* (pp. 167–174).

O'Mahony, M., Hurley, N., Kushmerick, N., & Silvestre, G. (2004a). Collaborative recommendation: A robustness analysis. *ACM Transactions on Internet Technology (TOIT)*, *4*(4), 344–377.

O'Mahony, M.P., Hurley, N.J., & Silvestre, G.C. (2004b). Efficient and secure collaborative filtering through intelligent neighbour selection. In *Proceedings of the 16th European conference on artificial intelligence* (pp. 383–387).

O'Mahony, M.P. (2004). *Towards robust and efficient automated collaborative filtering. PhD dissertation*: University College Dublin.

Pitsilis, G., & Marshall, L.F. (2004). *A model of trust derivation from evidence for use in recommendation systems*. Computing Science: University of Newcastle upon Tyne.

Sandvig, J.J., Mobasher, B., & Burke, R. (2007). Robustness of collaborative recommendation based on association rule mining. In *Proceedings of the 2007 ACM conference on Recommender systems* (pp. 105–112).

Tipping, M. (2001a). The relevance vector machine. *Advances in Neural Information Processing Systems*, *12*, 652–658.

Tipping, M.E. (2001b). Sparse Bayesian learning and the relevance vector machine. *The journal of machine learning research*, *1*, 211–244.

Weng, J., Miao, C., & Goh, A. (2006). Improving collaborative filtering with trust-based metrics. In *Proceedings of the 2006 ACM symposium on Applied computing* (pp. 1860–1864).

Williams, C.A., Mobasher, B., & Burke, R. (2007a). Defending recommender systems: detection of profile injection attacks. *Service Oriented Computing and Applications*, *1*(3), 157–170.

Williams, C.A., Mobasher, B., Burke, R., & Bhaumik, R. (2007b). Detecting profile injection attacks in collaborative filtering: a classification-based approach. In *Proceedings of the 8th Knowledge Discovery on the Web International Conference on Advances in Web Mining and Web Usage Analysis* (pp. 167–186).

Xu, H., Wu, X., & Li, X. (2009). Comparision study of Internet recommendation system. *Journal of Software*, *20*(2), 350–362.

Zhang, S., Ouyang, Y., Ford, J., & Makedon, F. (2006). Analysis of a low-dimensional linear model under recommendation attacks. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 517–524).