Information Sciences 505 (2019) 265–281

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Slanderous user detection with modified recurrent neural networks in recommender system

Yuanbo Xu[a], Yongjian Yang[a], Jiayu Han[a], En Wang[a,*], Jingci Ming[b], Hui Xiong[b]

[a] Jilin University, Changchun, Jilin, China
[b] Rutgers, the state university of New Jersey, NJ, USA

ARTICLE INFO

Article history:
Received 13 November 2018
Revised 22 July 2019
Accepted 24 July 2019
Available online 24 July 2019

Keywords:
Slanderous user detection
Recommender systems
Recurrent neural networks

ABSTRACT

We focus on how to tackle a unique multi-view unsupervised issue: slanderous user detection, with recurrent neural networks to benefit recommender systems. In real-world recommender systems, some consumers always give fake reviews and low ratings to the items they bought on purpose. In order to ensure their profits, these slanderous users make a semantic gap between their ratings and reviews to avoid detection, which makes slanderous user detection a more difficult problem. On some occasions, they give a false low rating with a positive review which confuse recommender systems, and vice versa. To address the above problem, in this paper, we propose a novel recommendation framework: Slanderous user Detection Recommender System (SDRS). In SDRS, we design a Hierarchical Dual-Attention recurrent Neural network (HDAN) with a modified GRU (mGRU) to compute an opinion level for reviews. Then a joint filtering method is proposed to catch the gap between ratings and reviews. With joint filtering, slanderous users can be detected and omitted. Finally, a modified non-negative matrix factorization is proposed to make recommendations. Extensive experiments are conducted in four datasets: Amazon, Yelp, Taobao, and Jingdong, in which the results demonstrate that our proposed method can detect slanderous users and make accurate recommendations in a uniform framework. Also, with slanderous user detection, some state-of-the-art recommendation systems can be benefited.

1. Introduction

Recommender systems have been widely applied to different domains in recent years [23,27]. Generally, recommender systems can predict consumers' preferences on different unconsumed commodities by analyzing their previous consumption history (reviews, ratings, *etc.*) [37]. Through this technology, merchants hope to make enormous profits by accurate recommendations, while consumers may enjoy a superior shopping experience at the same time, which can create a win-win situation.

However, some consumers with selfish intentions, namely *Slanderous Users*, are trying to make the personal gain by utilizing the vulnerabilities of recommender systems. As is well known, recommender systems can be considered as a technology to use the known information to predict the unknowns [7,38]. Correspondingly, slanderous users utilize fake information, ratings or reviews to confuse recommender systems and consumers. Because slanderous users are ordinary consumers

* Corresponding author.
E-mail address: wangen@jlu.edu.cn (E. Wang).

https://doi.org/10.1016/j.ins.2019.07.081
0020-0255/

whose interactions can cheat traditional abnormal user detection model [10], therefore they can keep giving fake information until it hurts the profit of merchants. Then, they will blackmail the merchants until getting paid. With the boosting of online business, this phenomenon is becoming more and more pervasive, leading to a lose-lose situation for both merchants and consumers.

To understand the details about how slanderous users utilize the vulnerabilities of recommender systems, we analyzed some real cases in an e-commerce website (Taobao). Basically, there are two kinds of recommender systems widely applied in real-scenarios: rating-based and review-based ones [4,15]. Correspondingly, there are two kinds of slanderous users: fake-rating and fake-review slanderous users. Both slanderous users consume some items sold by merchants online and give fake information on ratings and reviews. Fake-rating slanderous users always leave good reviews about the items, but give very low ratings. This interaction can severely hurt the credit of merchants because their online credits are based on the average ratings of consumed commodities. Meanwhile, rating-based recommender systems are severely confused by the fake ratings [5,8]. Fake-review slanderous users usually give good ratings but leave slanderous reviews. This interaction can hurt merchants because consumers will scan the reviews about the item before their consumption. Therefore, review-based recommender systems are badly affected by the fake reviews. In conclusion, both slanderous users can confuse recommender systems and hurt the profits of consumers and merchants [24].

Slanderous user detection is a new and challenging problem because 1) Most slanderous users are professional for a special purpose and can avoid traditional abnormal user detection. They do not give fake-rating and fake-review at the same time for the same merchant. So technically all the interactions of slanderous users are normal ones. It is difficult to label slanderous users from normal consumers by abnormal users detection, which forms a non-label, unsupervised problem. 2) To detect slanderous users, we need to analyze both ratings and reviews to find the real purpose of consumers, which could not be achieved by rating-based and review-based methods. It is a complicated multi-view problem based on ratings and reviews, and traditional abnormal user detection and recommender systems cannot solve it directly [18].

Along with this line, we treat slanderous user detection as an unsupervised learning problem. Taking a deep insight into the interactions slanderous users do, we notice that the core idea they utilize is that they do not give fake ratings and reviews at the same time, which means that there is an *opinion gap* between their reviews and ratings, while normal consumers express their opinions through both reviews and ratings without any opinion gap. Once we catch the opinion gap between reviews and ratings, slanderous users will be detected from normal consumers. Moreover, in order to catch the opinion gap, we need to utilize sentiment analysis on reviews and compare them with ratings. After slanderous user detection, we can improve the performance of recommender systems by dropping the fake information given by slanderous users.

To this end, in this paper, we propose a novel recommendation framework, **S**landerous users **D**etection **R**ecommender **S**ystem (SDRS), to tackle the slanderous user detection problem as a multi-view unsupervised problem and improve the performance of recommender system. In SDRS, we first utilize a Recurrent Neural Network based method to analyze the reviews. Then we calculate an opinion level for each review. We also design a Joint Filtering method to catch the opinion gap between ratings and reviews. Then we drop the fake information given by slanderous users and form a filtered user-item matrix. Finally, we employ a modified non-negative Matrix Factorization based method to make a recommendation. To the best of our knowledge, it is the first work to tackle the slanderous user detection as a multi-view unsupervised problem with ratings and reviews. Because our method is theoretical, different models for text sentiment analysis and recommendations can also be easily applied as modules of SDRS. We also evaluate our framework SDRS on two widely-used datasets: Yelp and Amazon and two self-collected real-world datasets: Taobao, JingDong. Extensive experiments demonstrate that SDRS can detect slanderous users from normal consumers and improve the performance of recommender systems.

This work's contributions can be summarized as follows:

- We present a multi-view unsupervised problem in existing recommender systems: slanderous user detection. To tackle this problem, we propose a novel recommendation framework: Slanderous user Detection Recommender System (SDRS), which can detect slanderous users and improve the performance of recommender systems. The idea of our framework is theoretical, so it can be easily applied to different recommender systems models.
- In SDRS, we design a Hierarchical Dual-Attention recurrent Neural network (HDAN) to analyze the text of reviews and calculate the opinion level. Especially, a modified GRU is applied in HDAN to improve the performance. Then a Joint Filtering method is proposed to catch the opinion gap between ratings and reviews and a modified non-negative matrix factorization model (MNMF) is employed to make recommendations.
- We conduct extensive experiments on real-world datasets, in which the encouraging results demonstrate that our proposed framework: 1) detects both types of slanderous users in a uniform framework with a stable performance. 2) benefits the state-of-the-art recommender system baselines.

The rest of this paper is organized as follows: In Section 2, we introduce the related work briefly. Some basic definitions of slanderous user detection and SDRS are introduced in Section 3. Then we introduce the framework of SDRS in Section 4 and the details in Section 5. In Section 6, we conduct experiments to evaluate our proposed model. Finally, we conclude our work in Section 7.

2. Related work

2.1. Abnormal user detection

As we define slanderous users in recommender systems, the slanderous user detection is a novel problem. As far as we know, there are few researches focusing on this problem, but we can treat our slanderous user detection problem as a special case of abnormal user detection, and there are some works in this area which can give us some inspirations. In e-commerce, various abnormal users (spammers, shilling group and frauds) have greatly damaged the system. First, [33] proposed a hybrid model to detect the spammers through users' profile and relations. To improve the framework, [11,14] explored spammer detection in big and sparse data. Shilling attacks harm the recommender system by injecting fake profile information of users and items. They cheat the recommendation model such as Collaborative Filtering, Matrix Factorization [29]. [43] proposed this attack type and gave a basic solution to tackle it. [29] proposed a convolutional network to tackle shilling attacks and improve the collaborative filtering. Frauds usually give fake reviews to hurt the profits of merchants. [1,32] also explored the fraud detection in large-scale dataset and real scenarios. The researches on these three types abnormal user detection usually focus on how to clustering users and find the fake information that abnormal users offered.

However, different from abnormal users mentioned above (spammers, shilling group and fraud), slanderous users are smarter and craftier. All the actions slanderous users take is well-behaved by the rules of e-commerce websites. Slanderous users utilize the drawbacks of abnormal detection, creating a gap between their ratings and reviews as we introduced in the previous section. Then they can make their own profits and harm the merchants. Basically, they are normal users so the existing abnormal user detection models may find it difficult to sove this problem. Our idea is to compute the reviews score, compare it with ratings and filter the slanderous users to benefit the recommender system models.

2.2. Text classification

Text classification is a basic technology to tackle the text in data mining area. There are many different types of text classification models that have been widely applied in different areas. In early time, text classification models were built based on probabilistic models. Then some researchers tried to apply traditional classification methods on text classification, such as SVM [9], TF-IDF [26]. Some researches focused on the power of latent vector presentation about expressing a text's theme and proposed labeled LDA [25]. Wang and Qian [31] combined SVM and LDA to achieve a balanced performance.

Recently, with the development of neural networks, many approaches based on neural networks have been proposed [17,20,41,42]. Most of them are based on convolutional neural networks [16,17,41] or recurrent neural networks [20,39,42]. For CNN, researchers of Kim [17] proposed a CNN model to do text classification, which included CNN-rand, CNN-multichannel. Then [41] improved the basic CNN model from a character view, which achieved a good classification result. Kim et al. [16] combined traditional MF with CNN and proposed a convolutional MF model on text classification. For RNN, Lai et al. [20] utilized the advantages of RNN to improve the performance, while [42] tried to consider the details of RNN about every character in documents. Yang et al. [39] applied dual-attention in RNN and achieved a relatively good result among different NN models.

Although there are many existing models, our proposed model is quite different: first, our model HDAN focuses on how to explicitly express users' true opinion on items by a sentiment score, instead of classification simply like other models did. Second, our model is a modified RNN with modified GRU (mGRU), which is designed for short reviews that users' opinions are consistent in them. Finally, text classification in our model is only a module, where its result should be input to other modules to make a recommendation.

2.3. Recommender system

Recommender systems with the neural network is becoming a hot research trend [3,6,12,22,36,40,45]. He et al. [12] utilized Multilayer perceptron (MLP) to design a network NCF to tackle implicit feedback recommendation problems. NCF is a rating-based model which can cover basic MF and CF and also achieve state-of-the-art performance. Moreover, Bai et al. [3] focused on the relations between neighbors and proposed a neural network-based recommender system. Also, some researchers try to combine neural models with traditional machine learning to make recommendations. Yang et al. [36] combined semi-supervised and neural network, bridged them and reinforced mutually. Yang et al. [38] proposed a novel concept: Serendipity and they utilized an MLP-based network to tackle the serendipity issues in recommender systems. Attention vectors are also employed by some researchers. [28] used local and global attention vectors to optimize user embedding in recommender systems. These models put more attention to the methodology of neural network itself rather than the applications in real scenarios, which also achieve a satisfying performance on various prefiltered datasets. They only validate their models on standard datasets like movielens or yelp but without the consideration of slanderous users in real-world scenario. Therefore, we propose SDRS in this paper, which is designed for slanderous user detection to improve the recommendations.

3. Preliminaries

In this section, we introduce some basic definitions, and we formulate the slanderous user detection problem with the following definitions.

3.1. Basic definitions

In recommender systems, we use U as the consumer set and I as the commodity set (in the following sections, user and item also mean consumer and commodity), $|U| = m, |I| = n$. R is the user-item rating matrix whose entry is ratings, r_{ui}. And T is the user-item review matrix whose entry is reviews t_{ui}. r_{ui} and t_{ui} are pair-occurred. $R, T \in \mathbb{R}^{m\times n}$. So we get U, I, R, T as the input of our method, where R, T can be treated as the interaction between users and items.

First, we need to define slanderous users in recommender systems. As we introduced before, there are two different kinds of slanderous users: fake-rating slanderous users and fake-review slanderous users. However, they share a similarity - there is a large opinion gao between the ratings and the reviews. In order to define slanderous users, we need to define slanderous interaction first.

Definition 1 Slanderous Interaction**.** Given a rating r_{ui} and a review t_{ui}, a slanderous interaction $d_{ui} = 1$ is the situation that $|or_{ui} - ot_{ui}| \geq \alpha$ and a normal interaction $d_{ui} = 0$ is that $|or_{ui} - ot_{ui}| < \alpha$.

or_{ui} is the opinion level of ratings, or_{ui} is the opinion level of reviews, d_{ui} is the slanderous interaction indicator and α is the threshold for opinion gap. Based on this, we could define slanderous user in the recommender system as follows:

Definition 2 Slanderous User**.** Given U, I, R, T, a slanderous user set is $\{u^s | \frac{\sum_{i\in I} |d_{ui}=1|}{\sum_{i\in I}(|d_{ui}=1|+|d_{ui}=0|)} \geq \beta, u \in U\}$, where β is a threshold of slanderous user detection. Note that even normal users make possible mistakes when they give ratings and reviews, so maybe there is a little portion of slanderous interactions. However, our model only wants to pick the professional slanderous users who make slanderous interactions and treat the historical interactions as ground truth. To avoid wrong detection, we need to utilize thresholds α and β. Moreover, in order to detect slanderous users, historical data is necessary as ground truth. For example, if there comes a new user, no persons or systems can tell whether he is a slanderous user without any ratings and reviews. To better tackle this kind of "cold start" detection, more side information (user attributes, item attributes, etc.) may be needed as a future work, which is not considered in this paper.

3.2. Problem definitions

In order to this multi-view unsupervised problem - slanderous user detection - we need to infer users' opinion level *or*, based on ratings *r* and *ot*, and reviews *t*. Then according to Definition 1, 2, we can get the slanderous user set. Hence, slanderous user detection can be treated as a two-phase process: slanderous user detection and recommendation. The definition of slanderous detection is shown as follows:

Problem Definition-Slanderous User Detection: Given U, I, R, T, slanderous user detection is a two-phase problem: 1) Find the slanderous users based on Definition 1, 2 and 2) Utilize the slanderous users to improve the performance of recommender system.

To solve the problem, firstly, we need to compute both *or* and *ot*. In this paper, we use a user-item rating *r* as rating opinion level *or* directly. Then we build a Recurrent Neural Network with modified GRU to analyze review *t* and output a review opinion level *ot*, which is in the same range of *or*. A Joint Filtering method is proposed to detect the slanderous users and try to improve the performance to a great extent. Finally, an MF-based recommendation model is employed to evaluate the performance. Some important notations are shown in Table 1.

Table 1
Notation list.

Notation	Description
U	user set
I	item set
R, T	rating/review set
m, n	number of users/items
r_{ui}	u's rating on item i
t_{ui}	u's review on item i
d_{ui}	slanderous interaction indicator
α	threshold for slanderous interaction
β	threshold for slanderous user
μ	weight for reviews' and ratings' opinion level
η	threshold for latent dimension selection in MNMF

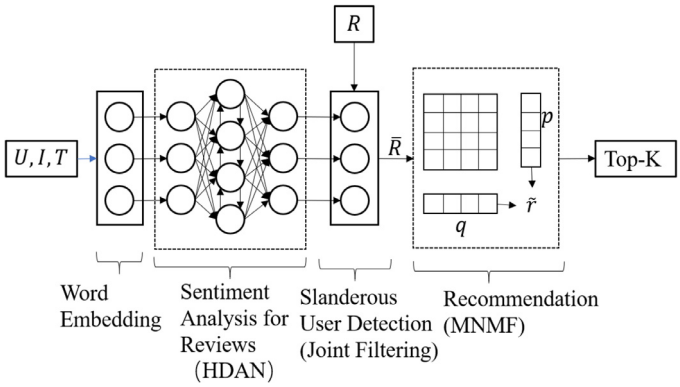


Fig. 1. Framework of SDRS.

4. Framework of SDRS

SDRS is composed of four modules: word embedding, sentiment analysis for reviews, slanderous user detection, and recommendations. First, we utilize word-embedding to map all the words in the reviews into vectors in word embedding. Then, we design a Hierarchical Dual-Attention recurrent Neural network (HDAN) to analyze users' review opinion levels *ot* in sentiment analysis of reviews. By comparing users' ratings r_{ui} and *ot*, we modify slanderous interactions and drop slanderous users in joint filtering, then build a filtered user-item matrix $\overline{R}$ as the input of the next module. Finally, a modified non-negative matrix factorization recommender system (MNMF) is proposed to utilize $\overline{R}$ for recommendations. Our framework is shown in Fig. 1.

Generally, this framework has two advantages for slanderous user detection: 1) SDRS utilizes the gap between users' reviews and ratings to detect slanderous interactions and users, which tackles with the non-label dataset and solves the unsupervised learning problem for this scenario. 2) The modules of SDRS are loose-coupling, which means that you can use other sentiment analysis model to replace HDAN or other state-of-the-art models to make recommendations. The joint filtering method can make sense between these modules and ensure the performance of this framework. This framework is flexible to be applied to different real-world scenarios.

5. Details of SDRS

In this section, we introduce the details of Word Embedding, Hierarchical Dual-Attention recurrent Neural network (HDAN), Joint Filtering and modified NMF (MNMF).

5.1. Word embedding

The first module of SDRS is a word embedding module, which is an effective method to find the relations between words. With *T* as the input, we employ Word2Vec [35], which is a rather mature tool to do word embedding. To simplify the problem, if the sentences in a review is longer than a threshold L_s, we drop the abundant sentences. Otherwise, we fill up the review with 0s to the length of L_s. Also, we set a threshold L_w for words in a sentence and do the same operation.

Because our work uses not only English reviews (Amazon, Yelp) but also Chinese reviews (Taobao, Jingdong), we borrow the idea of Li et al. [21] to modify traditional Word2Vec. After the word embedding, each word is transferred into a vector *w*.

5.2. Hierarchical dual attention RNN-HDAN

We design a hierarchical dual attention RNN (HDAN) to calculate the opinion level *ot* for each review. We input each sentence into W-level RNN with W-attention, then input each review into S-level RNN with S-attention. This structure is inspired by HAN [39]. However, our proposed HDAN utilizes a modified GRU instead of bidirectional GRU in HAN, which is more proper to analyze short reviews in our proposed SDRS. The structure of HDAN is shown in Fig. 2.

First, we introduce our modified GRU (mGRU). Traditional bidirectional GRU [2] utilizes a gating mechanism to hold memory about context without separate units. So does mGRU. There are two gates in mGRU: reset gate re_t and update gate ug_t. Both of them control how mGRU update the information in time *t*. At time *t*, GRU updates the information as follows:

$$h_t = (1 - ug_t) \odot h_{t-1} + ug_t \odot \widetilde{h}_{t-1}, \tag{1}$$

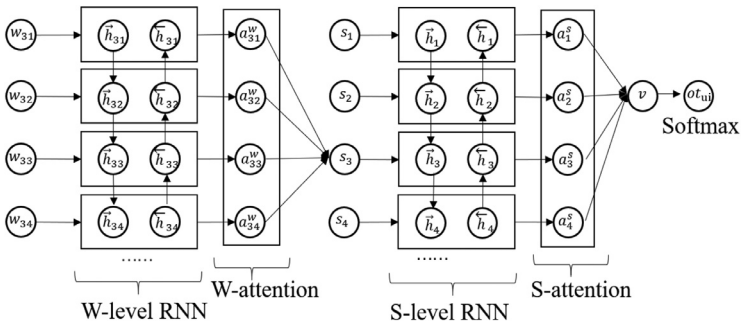


Fig. 2. Hierarchical dual attention RNN.

It's a linear function to combine former state h_{t-1} and temporary state $\widetilde{h}_{t-1}$, which is calculated by new sequence information. Traditional GRU calculate ug_t as follows:

$$ug_t = \sigma(W_{ug} y_t + U_{ug} h_{t-1} + b_{ug}), \tag{2}$$

where y_t is the current sequence of input.

However, as the sentences of review are usually short and the number of sentence is small in a review, we input $\widehat{y} = (y_{t-1}, y_t, y_{t+1})$ to replace y_t to catch more information in review. There are two advantages of this modification: 1) We can model users' dynamic opinions for each word of reviews. Traditional GRU is designed for long document classification, or voice recolonization, where the theme is changeable for each word in context. While in mGRU, we utilize $\widehat{y}$ to replace y_t to keep the opinions' consistency, which is proper for the scenarios. 2) The computation, no matter Back or Forward Propagation, can be simplified greatly, which can reduce the processing time. This is also the core improvement compared with [39]. With this modification, mGRU is more flexible and applicable in short texts, which is exactly the situation this paper is dealing with. So the calculation of ug_t in mGRU is shown as follows:

$$ug_t = \sigma(W_{ug}\widehat{y} + U_{ug} h_{t-1} + b_{ug}), \tag{3}$$

And temporary state $\widetilde{h}_t$ can be calculated as follows:

$$\widetilde{h}_{t-1} = \tanh(W_h \widehat{y} + re_t \odot (U_h h_{t-1}) + b_h), \tag{4}$$

where we also replace y_t by $\widehat{y}$. Here, reset gate re_t is the weight considering how much we should keep the former state. If $re_t = 1$, we need to keep the whole former state h_{t-1}. In mGRU, re_t is updated as follows:

$$re_t = \sigma(W_{re}\widehat{y} + U_{re} h_{t-1} + b_{re}), \tag{5}$$

After introducing mGRU, we also propose our dual-attention on word-level and sentence-level to make an accurate sentiment analysis and compute opinion level *ot*. We assume that a review contains L_s reviews, and each review contains L_w words. w_{it} stands for the tth word in the ith sentence and $t \in [1, L_w]$, $i \in [1, L_s]$.

To analyze sentiment in word level, we employ an embedding matrix W_e to map w_{it} into a vector y_{it}. As introduced before, mGRU is also a bidirectional unit, which has a forward function $\overrightarrow{mGRU}$ and a back forward function $\overleftarrow{mGRU}$:

$$y_{it} = W_e w_{it}, t \in [1, L_w],$$

$$\overrightarrow{h}_{it} = \overrightarrow{mGRU}(y_{it}), t \in [1, L_w],$$

$$\overleftarrow{h}_{it} = \overleftarrow{mGRU}(y_{it}), t \in [L_w, 1],$$

Then, we combine $\overrightarrow{h}_{it}$ and $\overleftarrow{h}_{it}$ as h_{it}: $h_{it} = [\overrightarrow{h}_{it}, \overleftarrow{h}_{it}]$, which contains all the sentiment information taking the word w_{it} as the center.

Note that not all the words play the similar role in a sentience, so we employ attention theory to compute different weights for each word. Obviously, a word's weight depends on the sentiment information h_{it}, and we utilize the following functions to calculate the attention weights:

$$c_{it}^w = \sigma(W_w h_{it} + b_w),$$

$$a_{it}^w = \frac{\exp((c_{it}^w)^T c^w)}{\sum_t \exp((c_{it}^w)^T c^w)},$$

$$s_i = \sum_t a_{it}^w h_{it},$$

where c_{it} can be treated as a hidden representation of h_{it} for the sentiment weight. c^w is a random vector which has the same dimension as c_{it} and is jointly learned in the training process.

After word-level, we move forward to sentence level. We do the same calculation in s-level RNN as w-level:

$$\overrightarrow{h}_i = \overrightarrow{mGRU}(y_i), i \in [1, L_s],$$

$$\overleftarrow{h}_i = \overleftarrow{mGRU}(y_i), i \in [L_s, 1],$$

Here, we also combine $\overrightarrow{h}_i$ and $\overleftarrow{h}_i$ as h_i: $h_i = [\overrightarrow{h}_i, \overleftarrow{h}_i]$, which contains all the sentiment information centering around sentence s_i.

Moreover, we need to compute the weight of different sentences to affect the review's sentiment, where an s-attention is applied as w-attention:

$$c_i^s = \sigma(W_s h_i + b_s),$$

$$a_i^s = \frac{\exp((c_i^s)^T c^s)}{\sum_i \exp((c_i^s)^T c^s)},$$

$$v = \sum_i a_i^w h_i,$$

where v is a review-level vector which contains almost all the information of the review. Then we utilize a softmax function to compute the opinion level ot:

$$ot = F(\text{softmax}(W_v v + b_v)), \tag{6}$$

where ot stands for the opinion level, F is a map function to normalize the opinion level into the same range of rating r. To train HDAN, we use user-item ratings to build our loss function shown as follows:

$$Loss = \sum_{r \in R} (r - ot)^2. \tag{7}$$

5.3. Joint filtering

The output of a well-trained HDAN is the opinion level for each review t_{ui}. Moreover, joint filtering is to utilize the opinion level and user-item rating to filter the slanderous interactions and slanderous users.

As concluded in the above, there is a gap between slanderous users' ratings and reviews. In details, it means that the opinion level ot is far apart from ratings r_{ui}. So in joint filtering, we filter the slanderous interaction, mark the slanderous interaction indicator d_{ui} with the following function:

$$\{d_{ui} | d_{ui} = 1, |ot_{ui} - r_{ui}| \geq \alpha; d_{ui} = 0, |ot_{ui} - r_{ui}| < \alpha\}, \tag{8}$$

With the indicator d_{ui}, we could build an indicator matrix R_{in}, $R_{in} \in \mathbb{R}^{m \times n}$. Then joint filtering tries to filter the slanderous users and drop them with the following function:

$$\left\{ u^s | \frac{\sum_{i \in I} |d_{ui} = 1|}{\sum_{i \in I} (|d_{ui} = 1| + |d_{ui} = 0|)} \geq \beta, u \in U \right\}, \tag{9}$$

By now, we have filtered slanderous interactions and slanderous users. To improve the performance of recommender systems, we need to utilize them to reform the original user-item matrix R: 1) For normal interactions of normal users, we utilize the following function to combine ratings and reviews linearly: $\bar{r}_{ui} = \mu r_{ui} + (1 - \mu) ot_{ui}$, where μ is a linear weight to balance the importance of ratings and reviews to make sure that both are effective for the opinion. 2) For slanderous interactions of normal users, we drop the ratings r_{ui} and reviews' opinion level ot_{ui}. 3) For slanderous users, we drop all the information about these users because of their unavailability and redundancy.

According to the joint filtering, we can leverage the results of slanderous user detection to build a filtered user-item matrix $\bar{R}$, whose entry is $\bar{r}$ and $\bar{R} \in \mathbb{R}^{(m - |u^s|) \times n}$, where $|u^s|$ is the number of slanderous users we have detected.

5.4. Modified NMF-MNMF

We propose a modified non-negative matrix factorization (MNMF) to make recommendations with filtered matrix $\bar{R}$. To make the demonstration simple, we use m instead of $(m - |u^s|)$ as the dimension of filtered matrix $\bar{R}$. Traditional non-negative matrix factorization models usually use the following factorization:

$$\bar{R} = PQ, P \in \mathbb{R}^{m \times k}, Q \in \mathbb{R}^{k \times n}, \tag{10}$$

where P can be treated as users' latent representation matrix, Q can be treated as items' latent representation matrix and k is the number of latent dimensions.

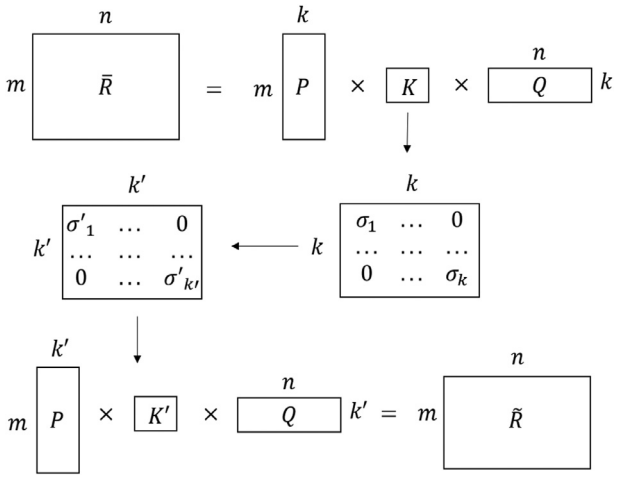


Fig. 3. The process of MNMF.

However, traditional NMF cannot reflect the importance of different latent dimensions. If the data is huge, it is difficult to decide the latent dimension k, as well as which latent dimensions should be utilized as the presentation of users and items. Therefore, we propose a modified non-negative matrix factorization (MNMF), which adds a weight diagonal matrix Σ to select latent dimension. MNMF can be formulated as follows:

$$\overline{R} = P\Sigma Q, P \in \mathbb{R}^{m\times k}, \Sigma \in \mathbb{R}^{k\times k}, Q \in \mathbb{R}^{k\times n}, \tag{11}$$

where Σ is a k-diagonal matrix whose entry is $\sigma_1, \sigma_2 \ldots, \sigma_k$. σ stands for the importance of latent dimension in user latent representation p in P and item latent representation q in Q. Then we can rerank the weights in Σ in descending order $\sigma'_1, \sigma'_2 \ldots, \sigma'_k$. Then we use top-$k'$ ($k' = \{k'|(\sigma'_1 + \sigma'_2 \ldots + \sigma'_{k'})/(\sigma_1 + \sigma_2 \ldots + \sigma_k) > \eta\}$) weights to rebuild a new k'-diagonal matrix Σ' whose entry is $\sigma'_1, \sigma'_2 \ldots, \sigma'_{k'}$. Finally, we select the most important top-k' dimensions in P and Q, which are consistent with Σ' and reconstruct the matrix. The whole process of MNMF is shown in Fig. 3.

After reconstructing the matrix $\widetilde{R}$, we can select the Top-k items from all the unrated items in the original matrix R of user u as the recommendations.

In general, SDRS can be divided into three procedures: Sentiment Analysis from Review (HDAN is proposed), slanderous user detection (Joint learning is proposed) and Recommendations (MNMF is proposed). All of these procedures can be easily applied jointly or partly in different scenarios and can be modified according to various requirements. The whole process of SDRS is shown in Algorithm 1.

Algorithm 1 Slanderous user Detection Recommender System (SDRS).

Input: user set U, item set I, original user-item matrix R, user-item reivew matrix T.
Output: Slanderous user list, Recommendation list.
Procedure 1: Sentiment Analysis from Review (HDAN):
1: Word Embedding for T.
2: Calculate users review opinion levels ot.
Procedure 2: slanderous user detection (Joint Filtering):
3: Use ot and original ratings r to detect slanderous users.
4: Build slanderous indicator matrix R_{in}.
5: **return** Slanderous user list.
6: Build filtered user-item rating matrix $\overline{R}$
Procedure 3: Recommendations (MNMF):
7: Use $\overline{R}$ instead of original matrix R as input.
8: **return** Top-k Recommendation list.

Table 2
The datasets' characteristics.

Dataset	Amazon	Yelp	Taobao	Jingdong
#user	30,759	45,980	10,121	8031
#item	16,515	11,537	9892	3025
#review	285,644	229,900	10,791	8310
#rating	285,644	229,900	49,053	25,152
Sparsity	0.051%	0.043%	0.049%	0.12%
Avg words /s	10.1	9.9	12.7	13.2
Avg words /r	104	130	65	70
Avg sentences /r	9.7	11.9	4.9	5.1
Avg reviews /u	9.29	5.00	1.06	1.03

6. Experimental results

6.1. Datasets

In order to validate the effectiveness of SDRS, we conduct abundant experiments on Amazon.com dataset[1] and Yelp for RecSys.[2] Amazon and Yelp datasets are two public datasets with abundant textual reviews. Moreover, we also collect two real-world datasets from Taobao[3] and Jindong[4] to validate slanderous user detection. The datasets all contain ratings ranging from 1 to 5, and we use 5-cross validation to divide the datasets, with 80% as training set, 10% as test set and 10% as validation set. The details of datasets are shown in Table 2 (/s, /r, /u mean per sentence/review/user). From Table 2, we can see that these datasets are extremely sparse. Note that not all the users in Taobao and Jingdong post reviews and ratings at the same time, so the number of ratings is larger than that of reviews in both datasets.

6.2. Metrics

Our proposed SDRS has three procedures, so we employ different metrics for different procedures.

For review sentiment analysis, we employ classification percentage and $A-err$ as the metric. Classification percentage means the accuracy of how the sentiment opinion level matches the rating. Because in SDRS, it should output a value rather than a catergory. So we utilize $A-err$, a proper MAE-style metric to measure the performance of the reviews' opinion level ot and ratings r, which is shown as follows:

$$A-err=\frac{\sqrt{\sum_{t_{ui}\in T}\left(ot_{ui}-r_{ui}\right)^2}}{|T||RR|},$$

where RR stands for rating range, which is 5-0=5 in this paper.

For review sentiment analysis, because the judgment of slanderous users is more subjective, which cannot be measured directly by some metrics. Therefore, we validate our slanderous user detection method from two aspects: 1) Directly, we employ several persons to judge whether the users we detected are slanderous, and make validations based on some websites, where the merchants will upload slanderous users list online.[5] 2) Indirectly, we filter the slanderous users we detected into recommender systems to see the change of performance. From both aspects, we can validate our slanderous user detection.

For recommendations, we employ two different metrics: Mean Squared Error (MSE) and Hitting Rate (HR). MSE is calculated as follows:

$$MSE@k=\frac{\sum_u\left(\widetilde{r}_{ui}-r_{ui}\right)^2/k}{|U|},$$

And HR is calculated as follows:

$$HR@k=\frac{\sum_u\left|I_u^{rec}\cap I_u^g\right|/k}{|U|},$$

where I_u^{rec} is the top-k recommendation set for user u, and I_u^g is the ground truth of user u.

[1] https://jmcauley.ucsd.edu/data/amazon.
[2] https://www.kaggle.com/c/yelp-recsys-2013.
[3] https://www.taobao.com.
[4] https://www.jd.com.
[5] http://www.taocece.com/.

6.3. Baselines

SDRS is composed of four modules, where word embedding is a mature tool and slanderous user detection is a novel issue with almost no benchmarks. So we compare SDRS with some sentiment analysis methods for reviews and recommendations:

For sentiment analysis for reviews: As HDAN we proposed is an RNN-based sentiment analysis model, we compare our model with two CNN-based models (CNN-rand and CNN-multichannel) and a state-of-the-art RNN-based model (HAN):

CNN-rand [17]: CNN-rand is a basic convolutional neural network-based method for sentence classification. It is a baseline model where all words are randomly initialized and then modified during training.

CNN-multichannel [17]: CNN-multichannel is an improved convolutional neural network-based method for sentence classification, a model with two sets of word vectors. CNN-multichannel is able to fine-tune one set of vectors while keeping the other static. Both channels are initialized with word2vec.

HAN [39]: HAN (Hierarchical Attention Network) is a hierarchical attention network for document classification. This model has a hierarchical structure that mirrors the hierarchical structure of documents and applied two levels of attention mechanisms on both the word and sentence-level. However, this model is designed for long document classification problem with standard GRU. Our HDAN is designed for short review sentiment analysis, with modified GRU (mGRU), which focuses more on our slanderous user detection problem.

For recommendations with explicit feedbacks: As MNMF we proposed is an MF-based recommendation model, we compare our model with one baseline (basic-CF), two MF-based models (NMF and Appro-SVD) and a state-of-the-art Neural Network based model (NCF):

basic-CF [19]: basic-CF (basic Collaborative Filtering) is a baseline method for recommendations. The idea of Collaboration Filtering is to find the relations about users and items, then recommend top-k nearest items to a specific user.

NMF [13]: NMF (Non-negative Matrix Factorization) is an MF-based recommendation technique for predicting the tastes of users in recommender systems based on collaborative filtering. This model is based on factorizing the rating matrix into two non-negative matrices whose components lie within the range [0, 1] with an understandable probabilistic meaning.

Appro-SVD [44]: Appro-SVD (Approximating the Singular Value Decomposition) is an incremental algorithm based on singular value decomposition (SVD) with good scalability, which combines the Incremental SVD algorithm with the Approximating the Singular Value Decomposition (ApproSVD) algorithm. The authors claimed that this method can achieve a state-of-the-art recommendation performance.

NCF [12] NCF (Neural Collaborative Filtering) is a neural-network-based recommendation method with explicit or implicit baselines. The author proposed a neural-network method with a vector combination function, which can overcome traditional drawbacks and achieve a stable recommendation result compared with other neural-network-based methods.

6.4. Parameter setting

We should set parameters in SDRS: In Word embedding, for English reviews (Amazon, Yelp), we set embedding dimension $L_s = 15, L_w = 12$; for Chinese reviews (Taobao, Jingdong), we set $L_s = 6, L_w = 15$. All the words are embedded into 20 dimensions as the latent space. In sentiment analysis for reviews, we utilize SGD to compute the weights, Adam optimizer with initialized learning rate 0.0001. In slanderous user detection, we set $\alpha = 3,\ \beta = 0.8,\ \mu = 0.5$ as initialization. All these parameters are determined through cross-validation to ensure the performance of SDRS. In recommendations, we set $\eta = 0.8,$ and make Top-3, Top-5 recommendations. To make the competition fair, we try our best to set the proper parameters for the baselines to achieve their best performance in our datasets [34].

6.5. Results and discussions

6.5.1. Sentiment analysis for reviews

We compare our proposed model HDAN with two CNN-based document classification models (CNN-rand and CNN-multichannel) and one RNN-based document classification method model (HAN). We also compare some basic sentiment classification methods such as BoW TFIDF [41], SVM [41] and LSTM [41]. The experimental results on all datasets are shown in Table 3.

Here we also compare HDAN with standard GRU (HDAN with GRU). From Table 3, we can see the improvement of our method regardless of data sizes and types (note that the reviews of Taobao and Jingdong are composed of Chinese characters while Amazon and Yelp are English). For smaller datasets, Taobao and Jingdong, the improvements on next-best method (HAN) are about 7.3% and 8.5%. While for larger datasets, the improvements are 6.5% and 4.2%.

Take a deep insight into the results, we notice that in terms of the data sparsity and size, some neural network methods, such as LSTM, CNN-rand, and CNN-multichannel, do not achieve a much better performance than traditional classification methods. For example, CNN-rand and CNN-multichannel cannot beat SVM in the case of Amazon datasets. The reason is that CNN-based model is relatively weak on catching the relations across the whole text.

Especially, when applied to Taobao and Jingdong, we notice that some traditional methods and neural-network model cannot achieve a satisfying result. The reason is that the relations between Chinese words are more various, complex and difficult to catch. Attention model is a popular method to catch the relations by calculating weights for different element

Table 3
Sentiment analysis results, %.

Dataset	Amazon	Yelp	Taobao	Jingdong
BoW TFIDF	55.4	60.1	31.1	33.3
SVM	56.1	62.1	33.2	34.1
LSTM	59.2	58.1	48.2	47.4
CNN-rand	54.5	58.6	44.7	45.8
CNN-multichannel	59.3	61.1	49.3	48.0
HAN	63.5	71.1	56.1	56.3
HDAN with GRU	63.3	71.2	56.5	56.9
HDAN with mGRU	**70.1**	**75.3**	**63.4**	**64.8**

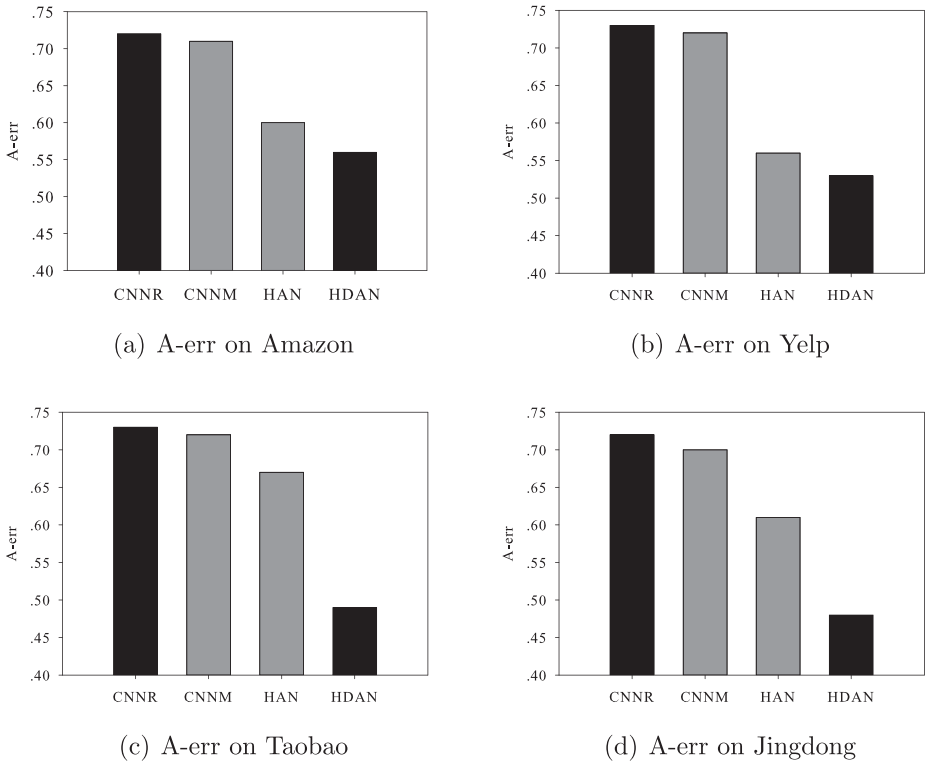


Fig. 4. A-err on different datasets with different models.

[30]. Compared with other models, RNN based models with attention model (HAN, HDAN with GRU and HDAN with mGRU) achieve a superior performance. Moreover, HAN and HDAN with GRU perform the same, while HDAN with mGRU is 10% better than them. This indicates that the mGRU that we modify in HDAN is proper for sentiment analysis problem in short texts.

To make a detailed comparison, we compute $A - err$ for CNN-rand (CNNR), CNN-multichannel (CNNM), HAN and HDAN. The results are shown in Fig. 4.

From the results, we can clearly see that our model overperforms the baselines. Moreover, we can see that on the standard datasets Amazon and Yelp, CNN-based models (CNNR, CNNM) do not achieve the same level performance as RNN-based models (HAN, HDAN), which demonstrates that RNN models are proper for this kind of sentiment classification problems. When comparing HAN with HDAN, we find that HDAN performs much better than HAN, especially in the case of Taobao and Jingdong. The reason is that users' opinions in reviews are relatively solid and consistent, and HDAN with mGRU can capture this character better than HAN with standard GRU for short reviews.

6.5.2. Slanderous user detection

To validate slanderous user detection, we use two different ways to define our ground truth on Taobao and Jingdong datasets: 1) We employ 20 people to manually tell the slanderous we filtered with our model, and give an error-rate analysis.

Table 4
Statistics of some filtered users in Taobao.

	ratings	reviews	5 star	4 star	3 star	2 star	1 star
user1	23	17	13	0	0	0	10
user2	46	22	4	1	0	0	41
user3	17	11	0	0	0	0	17
user4	25	22	3	0	0	0	22
user5	16	13	3	0	13	0	0
user6	5	2	0	0	0	0	5
user7	9	9	8	0	0	0	1

Table 5
Statistics of some filtered users in Taobao.

Dataset	Filtered users	Slanderous users	Detection accuracy
Taobao	37	32/avg	86.4%
Jingdong	17	14/avg	82.4%

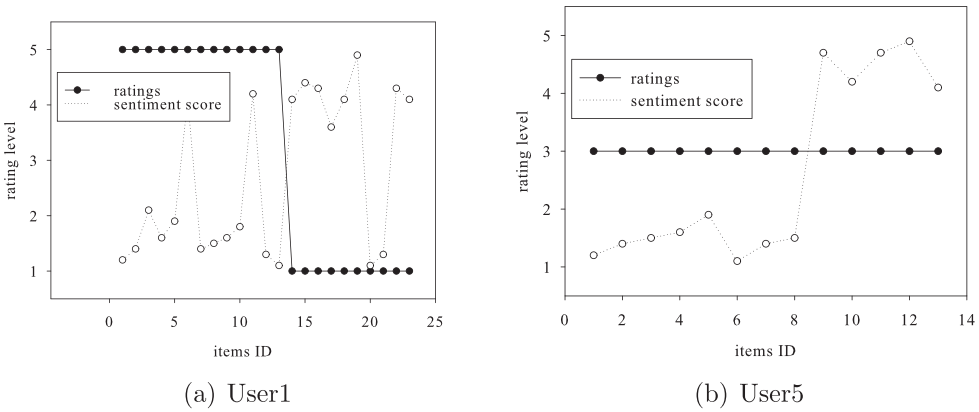


Fig. 5. Ratings and sentiment scores of filtered users.

2) We utilize an open slanderous users website[6] to question the users we filter. With both validations, we can see the effect of our model.

We filtered 37 users from Taobao dataset and Jingdong dataset 17. First, we give some statistics after the joint filtering module in Table 4 and Fig. 5:

From the statistics, we can see that these users gave balanced ratings and reviews to avoid traditional abnormal user detection. And we use user1 and user5 as examples. In Fig. 5, we can see that user1 gave balanced ratings (almost half-half for 5 stars and 1 star). However, his sentiment scores computed by our model are low when he rated 5 stars while high when rated 1 star. So did user5. Both two users are typical slanderous users we define in this paper, and our proposed model can filter them from massive users.

Then we employ 20 people to tell the users we filtered from the dataset. These people check the reviews and ratings the filtered users gave, and make their own judgments. The results are shown in Table 5.

The results show that our model can achieve an 80% accuracy to filter the slanderous users, and can replace manual actions to some extent. Finally, we check the ID of filtered slanderous users in the open slanderous website, and find 7 out of 37 on the website, which is shown in Fig. 6.

Considering that not all the slanderous users can be uploaded to the website, we think that the result is reasonable and trustworthy. From three different views of validations, we demonstrate that our proposed model can filter slanderous users, and complete the detection without manual supervision or labels. This experiment shows the potential to apply our model to real-world scenarios to tackle the slanderous user detection problem.

We also consider the parameter decisions for some important parameters, α and β, where α is the threshold of slanderous interaction, β is the threshold of slanderous user detection. We conduct validation for α and β on two real-world datasets Taobao and Jindong with Detection User Number (DUN) and Detection Accuracy (DA). The results are shown as follows:

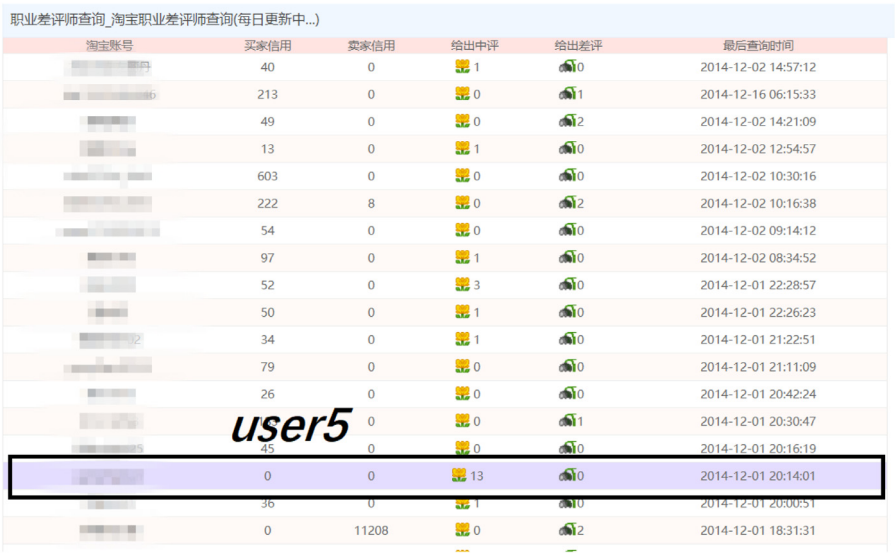


Fig. 6. Validation on open slanderous users report website.

Table 6
Parameter decision for α in Taobao and Jingdong.

Parameter α	DUN@Taobao	DA@Taobao	DUN@Jingdong	DA@Jingdong
1	192	16.7%	146	9.5%
2	103	31.0%	88	15.9%
3 (our choice)	**37**	**86.4%**	**17**	**82.4%**
4	3	66.6%	4	75%
5	0	–	0	–

Table 7
Parameter decision for β in Taobao and Jingdong.

Parameter β	DUN@Taobao	DA@Taobao	DUN@Jingdong	DA@Jingdong
0.2	321	9.9%	258	5.4%
0.4	163	19.6%	124	11.2%
0.6	143	23.8%	113	12.3%
0.8 (our choice)	**37**	**86.4%**	**17**	**82.4%**
1.0	7	85.7%	3	66.7%

From Table 6, we can see that our choice $\alpha = 3$ can achieve the best detection accuracy in both datasets. The smaller α may weaken the detection of SDRS for the reason that small opinion bias can usually occur in the reviews of not only slanderous users but normal users. Note that when we set $\alpha = 5$, our framework will lose the ability to filter the slanderous users because almost no user will give extreme polarized rating and review.

From Table 7, we can see that our choice $\beta = 0.8$ can achieve the best detection accuracy in both datasets. The smaller β may weaken the detection of SDRS for the reason that biased review and rating have a chance to happen in not only slanderous users but also normal users. Note that when we set $\beta = 1.0$, our framework can find less number but more accurate slanderous users (e.g, 1 slanderous user of 7 users in Taobao). Considering the real application, we need a wide detection rather than a narrow one, so we set $\beta = 0.8$.

6.5.3. Recommendation with explicit feedbacks

At first, we conduct experiments on four datasets to see the performance of our proposed MF recommendation method. We use the original user-item matrix as the input, and the results are shown in Tables 8, 9.

From Tables 8, 9, we can see that NCF and our proposed MNMF methods can achieve a same-level performance across the different datasets. However, NCF is a neural-network-based recommendation model, which needs more time to compute

[6] http://www.taocece.com/.

Table 8
Rating prediction for different models (MSE).

Dataset	basic-CF	NMF	Appro-SVM	NCF	MNMF
Amazon	0.913	0.871	0.856	0.857	**0.856**
Yelp	1.410	1.201	1.211	**1.198**	1.201
Taobao	1.512	1.341	1.674	**1.222**	**1.222**
Jingdong	1.672	1.555	1.54	1.332	**1.250**

Table 9
Top-5 for different models (HR).

Dataset	basic-CF	NMF	Appro-SVM	NCF	MNMF
Amazon	0.131	0.141	**0.222**	0.211	0.218
Yelp	0.172	0.177	**0.213**	0.200	**0.213**
Taobao	0.091	0.089	0.099	0.100	**0.185**
Jingdong	0.087	0.092	0.095	0.110	**0.166**

Table 10
Processing time for different models: T.

Dataset	basic-CF	NMF	Appro-SVM	NCF	MNMF
Amazon	70	81	132	270	150
Yelp	130	198	221	432	221
Taobao	46	50	66	212	71
Jingdong	47	66	74	211	73

Table 11
Effect of slanderous user detection.

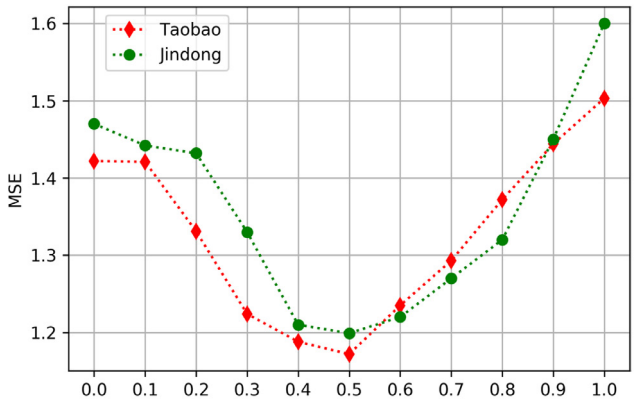
MSE	Taobao		Jingdong	
	R	$\overline{R}$	R	$\overline{R}$
basic-CF	1.512	**1.503(+0.5%)**	1.672	**1.60+(4.3%)**
NMF	1.341	**1.333(+0.5%)**	1.555	**1.542(+0.8%)**
Appro-SVM	1.674	**1.670(+0.2%)**	1.54	**1.522(+1.1%)**
NCF	1.222	**1.112(+9.0%)**	1.332	**1.120(+15.9%)**
MNMF	1.222	**1.172(+4.1%)**	1.250	**1.199(+4.0%)**
HR				
basic-CF	0.091	**0.101(+10.9%)**	0.087	**0.090(+3.4%)**
NMF	0.089	**0.100(+12.3%)**	0.092	**0.099(+7.6%)**
Appro-SVM	0.099	**0.118(+19.1%)**	0.095	**0.101(+6.3%)**
NCF	0.100	**0.172(+72%)**	0.110	**0.194(+89.4%)**
MNMF	0.185	**0.197(+6.4%)**	0.166	**0.197(+18.6%)**

the parameters than our proposed model does, especially in real-world unbalanced datasets Taobao and Jingdong (shown in Table 10). Under the consideration of time and effect, we think our model is more proper to be applied to those recommendation scenarios (Fig. 7). At last, we conduct experiments to show the effect of slanderous user detection, with original user-item matrix R and filtered user-item matrix $\overline{R}$ as input respectively. The results are shown in Table 11.

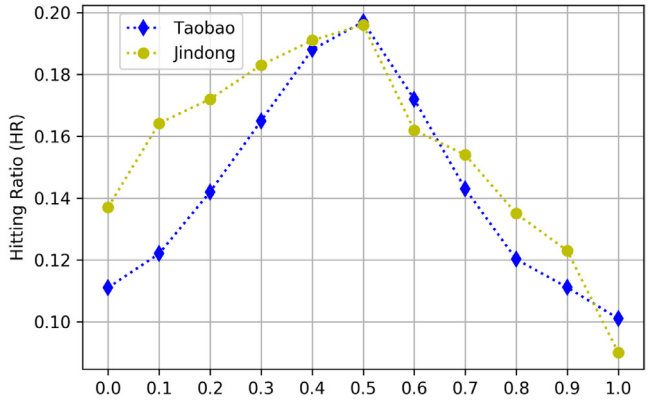
From the results, we can see that filtered matrix $\overline{R}$ can benefit the existing recommendation models, especially NCF, the neural-network-based model. The reason is that the neural network is easily affected by outliers, especially the slanderous users in this scenario. Note that the improvements on basic-CF and NMF are not as effective as MNMF, which means that our proposed method can maximize the effect of slanderous user detection and achieve a state-of-the-art recommendation performance.

To decide the important parameter μ in recommendations, we use MSE and HR as metrics, and Taobao and Jindong as datasets. The results are shown as follows:

From the results, we find that when we set $\mu = 0.5$, where we take a balance between ratings and opinion level, SDRS can achieve the best performance with MSE and HR in both datasets. Note that when we set $\mu = 1.0$, our SDRS fades to a basic CF model, which loses the advantage of SDRS.



(a) MSE with μ



(b) Hitting Ratio with μ

Fig. 7. Recommendation parameter decision for μ.

7. Conclusion

First, we introduced a new phenomenon in existing recommender systems: slanderous users, which can cause great damage to teh recommender system. Then we analyzed this phenomenon, built a multi-view unsupervised problem and proposed a novel recommender system framework, SDRS, to tackle this problem. SDRS utilizes only ratings and reviews to detect slanderous users without any other side information, which makes it easy and useful to be applied to different scenarios. Moreover, the framework of SDRS is convenient to be modified and readjusted for developers because all the modules of SDRS (Word Embedding, HDAN, etc.) are independent. Experiments on several real-world datasets also demonstrated that the efficiency and accuracy of SDRS are better than some state-of-the-art baselines.

In the future, we need to consider how to make SDRS more efficient and easier to be applied. We believe the theory of multi-task should be a good extension. Since slanderous user detection is a two-phase problem, it is suitable for multi-task learning. Moreover, the co-training theory can also be another exciting attempt to optimize our framework to tackle the cold start and data sparsity issues with more side information.

Declaration of Competing Interest

We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work, there is no professional or other personal interest of any nature or kind in any product, service and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled, “Slanderous user detection with modified recurrent neural networks in recommender system”.

Acknowledgment

This work is supported by the National Natural Science Foundation of China under Grant No. 61772230, 61773361, U1836206 and Natural Science Foundation of China for Young Scholars No. 61702215, China Postdoctoral Science Foundation No. 2017M611322 and 2018T110247.

References

[1] L. Akoglu, R. Chandy, C. Faloutsos, Opinion fraud detection in online reviews by network effects., ICWSM 13 (2013) 2–11.
[2] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, (2014) arXiv:1409.0473.
[3] T. Bai, J.-R. Wen, J. Zhang, W.X. Zhao, A neural collaborative filtering model with interaction-based neighborhood, in: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017, pp. 1979–1982.
[4] Y. Cao, X. Wang, X. He, Z. Hu, T. Chua, Unifying knowledge graph learning and recommendation: towards a better understanding of user preferences, in: The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13–17, 2019, 2019, pp. 151–161, doi:10.1145/3308558.3313705.
[5] P. Chamoso, A. Rivas, S. Rodríguez, J. Bajo, Relationship recommender system in a business and employment-oriented social network, Inf. Sci. 433–434 (2018) 204–220, doi:10.1016/j.ins.2017.12.050.
[6] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, T.-S. Chua, Attentive collaborative filtering: multimedia recommendation with item-and component-level attention, in: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2017, pp. 335–344.
[7] L. Chen, Y. Yang, N. Wang, K. Yang, Q. Yuan, How serendipity improves user satisfaction with recommendations? A large-scale user evaluation, in: The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13–17, 2019, 2019, pp. 240–250, doi:10.1145/3308558.3313469.
[8] L.M. de Campos, J.M. Fernández-Luna, J.F. Huete, L. Redondo-Expósito, Positive unlabeled learning for building recommender systems in a parliamentary setting, Inf. Sci. 433–434 (2018) 221–232, doi:10.1016/j.ins.2017.12.046.
[9] G. Forman, BNS feature scaling: an improved representation over TF-IDF for SVM text classification, in: Proceedings of the 17th ACM Conference on Information and Knowledge Management, 2008, pp. 263–270.
[10] A. Gogna, A. Majumdar, DiABLO: optimization based design for improving diversity in recommender system, Inf. Sci. 378 (2017) 59–74, doi:10.1016/j.ins.2016.10.043.
[11] B. Guo, H. Wang, Z. Yu, Y. Sun, Detecting spammers in e-commerce website via spectrum features of user relation graph, in: Advanced Cloud and Big Data (CBD), 2017 5th International Conference on, 2017, pp. 324–330.
[12] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, T.-S. Chua, Neural collaborative filtering, in: Proceedings of the 26th International Conference on World Wide Web, 2017, pp. 173–182.
[13] A. Hernando, J. Bobadilla, F. Ortega, A non negative matrix factorization for collaborative filtering recommender systems based on a Bayesian probabilistic model, Knowl.-Based Syst. 97 (2016) 188–202.
[14] T.V.R. Himabindu, V. Padmanabhan, A.K. Pujari, Conformal matrix factorization based recommender system, Inf. Sci. 467 (2018) 685–707, doi:10.1016/j.ins.2018.04.004.
[15] D.H. Kim, C. Park, J. Oh, H. Yu, Deep hybrid recommender systems via exploiting document context and statistics of items, Inf. Sci. 417 (2017) 72–87, doi:10.1016/j.ins.2017.06.026.
[16] D. Kim, C. Park, J. Oh, S. Lee, H. Yu, Convolutional matrix factorization for document context-aware recommendation, in: Proceedings of the 10th ACM Conference on Recommender Systems, 2016, pp. 233–240.
[17] Y. Kim, Convolutional neural networks for sentence classification, (2014) arXiv:1408.5882.
[18] A. Kleinerman, A. Rosenfeld, F. Ricci, S. Kraus, Optimally balancing receiver and recommended users' importance in reciprocal recommender systems, in: Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2–7, 2018, 2018, pp. 131–139, doi:10.1145/3240323.3240349.
[19] Y. Koren, R. Bell, Advances in Collaborative Filtering, Springer, 2015, pp. 77–118.
[20] S. Lai, L. Xu, K. Liu, J. Zhao, Recurrent convolutional neural networks for text classification., AAAI 333 (2015) 2267–2273.
[21] Y. Li, W. Li, F. Sun, S. Li, Component-enhanced chinese character embeddings, (2015) arXiv:1508.06669.
[22] J. Lian, F. Zhang, X. Xie, G. Sun, CCCFNeT: a content-boosted collaborative filtering neural network for cross domain recommender systems, in: Proceedings of the 26th International Conference on World Wide Web Companion, 2017, pp. 817–818.
[23] E.R. Núñez-Valdéz, D. Quintana, R.G. Crespo, P. Isasi, E. Herrera-Viedma, A recommender system based on implicit feedback for selective dissemination of ebooks, Inf. Sci. 467 (2018) 87–98, doi:10.1016/j.ins.2018.07.068.
[24] E.R. Núñez-Valdéz, D. Quintana, R.G. Crespo, P. Isasi, E. Herrera-Viedma, A recommender system based on implicit feedback for selective dissemination of ebooks, Inf. Sci. 467 (2018) 87–98, doi:10.1016/j.ins.2018.07.068.
[25] D. Ramage, D. Hall, R. Nallapati, C.D. Manning, Labeled LDA: a supervised topic model for credit attribution in multi-labeled corpora, in: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1, 2009, pp. 248–256.
[26] J. Ramos, et al., Using TF-IDF to determine word relevance in document queries, in: Proceedings of the First Instructional Conference on Machine Learning, vol. 242, 2003, pp. 133–142.
[27] F. Ricci, L. Rokach, B. Shapira, Recommender Systems: Introduction and Challenges, Springer, 2015, pp. 1–34.
[28] S. Seo, J. Huang, H. Yang, Y. Liu, Interpretable convolutional neural networks with dual local and global attention for review rating prediction, in: Proceedings of the 11th ACM Conference on Recommender Systems, 2017, pp. 297–305.
[29] C. Tong, X. Yin, J. Li, T. Zhu, R. Lv, L. Sun, J.J. Rodrigues, A shilling attack detector based on convolutional neural network for collaborative recommender system in social aware network, Comput. J. 61 (7) (2018) 949–958.
[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. (2017) 5998–6008.
[31] Z. Wang, X. Qian, Text categorization based on LDA and SVM, in: Computer Science and Software Engineering, 2008 International Conference on, vol. 1, 2008, pp. 674–677.
[32] H. Weng, Z. Li, S. Ji, C. Chu, H. Lu, T. Du, Q. He, Online e-commerce fraud: a large-scale detection and analysis, in: 2018 IEEE 34th International Conference on Data Engineering (ICDE), 2018, pp. 1435–1440.
[33] Z. Wu, Y. Wang, Y. Wang, J. Wu, J. Cao, L. Zhang, Spammers detection from product reviews: a hybrid model, in: Data Mining (ICDM), 2015 IEEE International Conference on, 2015, pp. 1039–1044.
[34] Y. Xu, Y. Yang, J. Han, E. Wang, F. Zhuang, J. Yang, H. Xiong, NeuO: exploiting the sentimental bias between ratings and reviews with neural networks, Neural Netw. 111 (2019) 77–88.
[35] B. Xue, C. Fu, Z. Shaobin, A study on sentiment computing and classification of Sina Weibo with Word2vec, in: Big Data (BigData Congress), 2014 IEEE International Congress on, 2014, pp. 358–363.
[36] C. Yang, L. Bai, C. Zhang, Q. Yuan, J. Han, Bridging collaborative filtering and semi-supervised learning: a neural approach for poi recommendation, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017, pp. 1245–1254.
[37] X. Yang, Y. Guo, Y. Liu, H. Steck, A survey of collaborative filtering based social recommender systems, Comput. Commun. 41 (2014) 1–10.
[38] Y. Yang, Y. Xu, E. Wang, J. Han, Z. Yu, Improving existing collaborative filtering recommendations via serendipity-based algorithm, IEEE Trans. Multimed. 20 (7) (2018) 1888–1900, doi:10.1109/TMM.2017.2779043.

[39] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, E. Hovy, Hierarchical attention networks for document classification, in: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016, pp. 1480–1489.
[40] H. Ying, F. Zhuang, F. Zhang, Y. Liu, G. Xu, X. Xie, H. Xiong, J. Wu, Sequential recommender system based on hierarchical attention networks, in: Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI-18), 2018.
[41] X. Zhang, J. Zhao, Y. LeCun, Character-level convolutional networks for text classification, Adv. Neural Inf. Process. Syst. (2015) 649–657.
[42] C. Zhou, C. Sun, Z. Liu, F. Lau, A C-LSTM neural network for text classification, (2015a) arXiv:1511.08630.
[43] W. Zhou, J. Wen, Y.S. Koh, Q. Xiong, M. Gao, G. Dobbie, S. Alam, Shilling attacks detection in recommender systems based on target item analysis, PloS one 10 (7) (2015).
[44] X. Zhou, J. He, G. Huang, Y. Zhang, SVD-based incremental approaches for recommender systems, J. Comput. Syst. Sci. 81 (4) (2015) 717–733.
[45] F. Zhuang, D. Luo, N.J. Yuan, X. Xie, Q. He, Representation learning with pair-wise constraints for collaborative ranking, WSDM '17 Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, 2017, pp. 567–575. 10.1145/3018661.3018720