

Article

# FSCR: A Deep Social Recommendation Model for Misleading Information

Depeng Zhang , Hongchen Wu  and Feng Yang \* 

School of Information Science and Engineering, Shandong Normal University, Jinan 250014, China; 2018020917@stu.sdnu.cn (D.Z.); wuhongchen@sdnu.edu.cn (H.W.)

\* Correspondence: yangfeng@sdnu.edu.cn

**Abstract:** The popularity of intelligent terminals and a variety of applications have led to the explosive growth of information on the Internet. Some of the information is real, some is not real, and may mislead people's behaviors. Misleading information refers to false information made up by some malicious marketer to create panic and seek benefits. In particular, when emergency events break out, many users may be misled by the misleading information on the Internet, which further leads them to buy things that are not in line with their actual needs. We call this kind of human activity 'emergency consumption', which not only fails to reflect users' true interests but also causes the phenomenon of user preference deviation, and thus lowers the accuracy of the personal recommender system. Although traditional recommendation models have proven useful in capturing users' general interests from user-item interaction records, learning to predict user interest accurately is still a challenging problem due to the uncertainty inherent in user behavior and the limited information provided by user-item interaction records. In addition, to deal with the misleading information, we divide user information into two types, namely explicit preference information (explicit comments or ratings) and user side information (which can show users' real interests and will not be easily affected by misleading information), and then we create a deep social recommendation model which fuses user side information called FSCR. The FSCR model is significantly better than existing baseline models in terms of rating prediction and system robustness, especially in the face of misleading information; it can effectively identify the misleading users and complete the task of rating prediction well.

**Keywords:** recommender system; rating predicts; deep learning; misleading information



**Citation:** Zhang, D.; Wu, H.; Yang, F. FSCR: A Deep Social Recommendation Model for Misleading Information. *Information* **2021**, *12*, 37. <https://doi.org/10.3390/info12010037>

Received: 11 December 2020

Accepted: 9 January 2021

Published: 17 January 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

In the 21st century, the Internet is closely related to the basic life of users. Humans use different platforms and applications to meet their different needs, such as shopping, entertainment, education, communication. These activities generate a lot of data, which can be used by various industries. For example, in [1], Vincenzo Moscato et al. describe a novel music recommendation technique based on the identification of personality traits, moods and emotions of a single user. In addition, in [2], Flora Amato et al. propose an original summarization technique of social media content applied to Online Social Networks (OSNs) for multimedia story creation.

The recommender system aims to solve the problem of information overload and help users get the best choice from a variety of choices [3]. The key idea of the recommender system is to predict the rating of a group of unrated items according to user historical behavior, and then select personalized recommendation from the items with the top predicted ratings [4].

To attract users, websites and apps need to be increasingly accurate with recommender systems [5–7]. In this case, the prediction ratings of the recommender system should be as close to the true ratings as possible. However, misleading information will reduce

the accuracy of the recommendation results, and then affect customer satisfaction. This is because a recommender system cannot distinguish which information is misleading information, and user/item representations are learned from such misleading information, so the representations learned are unreliable.

We obtain user preference information not only from user–item interaction but also from user side information and user trust information. A normal user’s behavior should be consistent and predictable [8]; even when the user’s explicit ratings do not match the user’s usual preference, the implicit preference in the user side information and user trust information is still relatively accurate. When learning the representation of users, we not only consider the historical rating record of users but also take into account the side information of users. By combining the explicit rating and social features of users, we can improve the robustness of the recommender system and reduce emergency consumption caused by misleading information.

We summarize the main contributions of our work as follows.

1. We introduce and analyze the emergency consumption problem caused by misleading information. To solve this problem, we optimize the recommendation model by combining user side information.
2. We introduce the FSCR model, which fuses user side information, historical preference features, and social trust features to build the user model and make recommendations.
3. We introduce a misleading information detection mechanism in our model to deal with misleading information.
4. We conduct comprehensive experiments in multiple real-world datasets to show the effectiveness and the robustness of the FSCR model.

The remainder of this paper is organized as follows. We provide some related work in Section 2, and then show our model in Section 3. We will introduce the experimental results in Section 4. Finally, we summarize this paper and discuss future directions in Section 5.

## 2. Related Work

We divide the related work into four parts. The first part briefly introduces the recommender system, the second part introduces the influence of misleading information dissemination, the third part introduces the factors that promote the dissemination of misleading information, and the last part introduces the methods of dealing with misleading information.

### 2.1. Recommender System

By analyzing user behaviors and modeling their personalized needs, recommender system can recommend some long-tail items to users and help them find the items they want but are difficult to be found.

There are three common ways to connect users and items: content-based recommendation, collaborative filtering, and social recommendation. Content-based recommendation is defined as recommending items to users that are similar to what they liked in the past [9,10]. Collaborative filtering finds users with similar interests in user groups by analyzing user interests, extracts the features of these similar users, and finally recommends items to users through these features [11].

Social recommendation uses user social behavior data to better model a user profile and user–item connection function [12]. With the increasing number of mobile phone users, social recommendation has become the most popular way to connect users and items. The general definition of the social recommendation model [13] is as follows:

$$Y = F(V, U, T) \quad (1)$$

$Y$  is predicted score or recommendation list;  $F$  represents the model or method used in the recommendation process;  $U$  represents the user attribute information;  $T$  represents the user social trust information; and  $V$  represents the user–item interaction information.

## 2.2. *The Harm of Misleading Information*

In recent years, misleading information aroused widespread concern in society. The growing concern largely from the widespread impact of misleading information on information security and public opinion [14].

Misleading information is widely distributed, harmful, and difficult to identify. On the personal level, misleading information not only misleads people's judgment but also makes it difficult for people to distinguish right from wrong [15]. Under the wrong guidance of misleading information, some people are easy to believe the misleading information and even forward it to their relatives and friends.

At the social level, misleading information can even influence public opinion by controlling the emotions of voters, thus influencing the outcome of national elections. Moreover, misleading information hurts the public interest of the country and affect the social order [16]. Moreover, misleading information often appears around emergencies, such as the earthquake in Japan in 2011 [17], hurricane Sandy in 2012 [18] and COVID-19 in 2019 [19], aiming to cause more panic and confusion.

## 2.3. *Promoters of Misleading Information Dissemination*

We will explain the generation and dissemination of misleading information from the perspective of individuals and society.

### 2.3.1. Individuals

Users cannot identify the misleading information accurately, which leads to users constantly sharing and believing misleading information on social media. The main reason for this phenomenon is that users generally believe the information they like. In other words, objective facts are not as powerful as personal emotions in guiding user interests [4].

### 2.3.2. Society

Although social networks help us to realize real-time information sharing, they also bring convenience for the spread of misleading information, accelerate the speed of false information dissemination, and expand the dissemination scope of misleading information.

The nature of information sharing on social media and online platforms provides additional help for the dissemination of misleading information, we commonly call it echo chamber effect [8]. Users always find and share information that is consistent with their views, while platform algorithms recommend items that suit personal preferences, as well as friends with similar preferences. The homogeneity of friends and the personalization of algorithm will make users less exposed to conflicting views. Moreover, the features of social networks make the spread of misleading information more complex, which makes misleading information affect numerous users in a short time [20].

## 2.4. *How to Deal with Misleading Information*

There are two common ways to reduce the influence of misleading information. One is to detect and prevent misleading information directly, the other is to increase the robustness of the system.

### 2.4.1. Detection of Misleading Information

The N-gram approach was one of the effective methods to detect misleading information. In [21,22], the authors used the N-gram method to analyze deceptive comments created by the crowdsourcing workers of Amazon: false positive comments about hotels [22] and false negative comments about hotels [21]. Research data showed that fake reviews contain fewer spatial words (location, floor, small) because the individuals had not

actually experienced hotels. In addition, they also found that positive emotional words are exaggerated in false positive reviews compared with real reviews. In the false negative comment, the negative emotion word also appears similar to exaggeration.

Deep learning method [23,24] could alleviate the shortcomings of the method based on language analysis by automatic feature extraction. It could extract both simple features and some hard-to-specify features. In [25], Wang et al. proposed a content-based false news detection using convolutional neural network. Compared with the traditional baseline model, they observed that the CNN model has better detection accuracy.

As the misleading information becomes diversified and intelligent, the identification of misleading information becomes more and more difficult, and it also consumes a lot of time and resources.

#### 2.4.2. Improve the Robustness of the System

To improve the robustness of the recommendation model, He et al. proposed a new optimization framework: adversarial personalized ranking (APR) [26]. APR improved the pairwise sorting method of BPR [27] through confrontation training, and maximizing the effective function of BPR. Experiments on several datasets show the effectiveness of APR model.

Users always post pictures or micro-videos on social media to interact with their friends. These multimedia contents contain rich information, which can show user preferences and provide opportunities for improving the recommender system [28]. Tang et al. found that the existing multimedia recommendation model lacks robustness. To solve this problem, they proposed an advanced multimedia recommendation (AMR) [29], which could enhance the robustness of the model against misleading information and reduced the disturbance of misleading information on the recommendation model. Then, machine learning [30–32] and generate countermeasure network [33–35] are also used to improve the robustness of the recommender system and deal with misleading information.

Traditional recommendation models usually improve the robustness of the recommender system by adding interference manually. However, in a recommender system, each user is not an isolated point. The user's behavior will affect the user's direct friends and users with similar interests. Therefore, adding disturbance terms to the training-set to improve the robustness of the recommender system will result in unsatisfactory training results. This is because the training of the recommendation model is very complicated. When learning user representations, the recommendation model should not only consider the users' historical behaviors but also consider the behaviors of the users' friends and users with similar interests. After adding a disturbance term, the learned user representation is often affected. In other words, the learned user representation deviates user interest and affects the accuracy of the recommendation.

In this paper, we propose a robust rating prediction system based on multi-domain information fusion. In particular, we jointly consider the user's historical behavior and user side information to predict the user's unrated items. In addition, we also embed a misleading user identification mechanism in our model to deal with the misleading information. Finally, we get the final prediction rating by aggregating the learned user representation and item representation.

### 3. Model Implementation

This section elaborates our proposed model. We first show the definition of the problem, and then introduce the various components of our model.

#### 3.1. Problem Definition

There are three matrices: user–item matrix, user feature matrix, user interaction matrix. The user–item matrix  $V \in R_{n \times t}$  is the rating matrix representing the user rating of items purchased or consumed;  $n$  is the number of users; and  $t$  is the number of items; the user feature matrix  $U \in R_{n \times s}$  ( $U$  is an irregular matrix because the user's side information tends

to be described differently depending on the type (gender is char, age is int, etc.);  $s$  represents the type of side information of the user;  $T \in R_{n \times n}$  represents the user interaction matrix; and if  $R_{i \times j} = 1$ , it represents that user  $i$  trusts or follows user  $j$ . Our goal is to embed the user's side information into the deep social recommendation model [29–31] to optimize user modeling, and then predict the user's unrated items using the three types of available information. In general, it is still necessary to learn the user's low-dimensional representation and item embeddings. We specify the symbol representations used in this paper (in Table 1).

**Table 1.** Symbol representations.

Symbol	Representation
$R_{n \times t}$	rating matrix
$R_{n \times s}$	user side information
$R_{n \times n}$	user trust matrix
$R_{n \times k}$	user feature matrix
$R_{t \times k}$	item feature matrix
$n$	the number of users
$t$	the number of items
$s$	the types of user features
$k$	the characteristic dimension
$m$	the number of items that interact with the user
$U_s$	user's side information embedding
$w$	neural network weight
$b$	bias
$\sigma$	the Relu activation function
$y_p$	the prediction rating
$y_a$	the actual rating
$M_u$	the mean square of all rating prediction errors on the interaction item

### 3.2. FSCR Model Architecture

The structure of FSCR (in Figure 1) includes five main parts: the embedding layer, the MF layer, the social influence diffusion layer, the FC layer, and the prediction layer. To be specific, by obtaining relevant input, the embedding layer outputs the user's side information embedding. The MF layer uses the user–item interaction matrix as input to output the user's hidden preference embedding and item feature embedding. In the social influence diffusion layer, we design a model which can effectively measure how user opinions spread in social networks to solve the social embeddedness problem exposed by users in social network. The FC layer fuses user side information embedding, hiding preference embedding, and social embedding to get the complete user feature profile. Finally, the prediction layer predicts the score of each unobserved user–item pair.

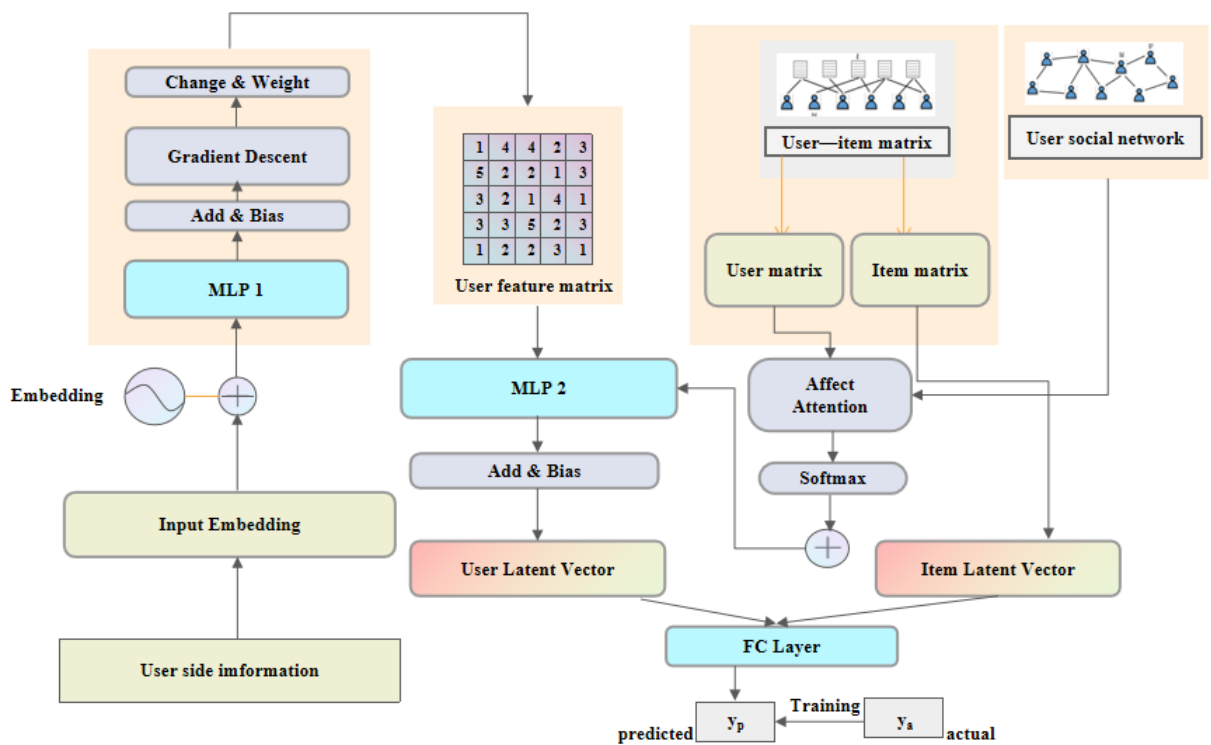


Figure 1. FSCR Model Architecture.

### 3.3. Embedding Layer

Here we take the movielens dataset (<https://grouplens.org/datasets/movielens/>) as an example. In the movielens dataset, there are three types of user side information: age, gender, and job. After adding the user id, we have four types of embedded information. Afterwards, we perform the embedding operation and get id-embed, gender-embed, age-embed, and job-embed, to express convenience we use  $x_1; x_2; x_3; x_4$  to replace. Finally, we get user’s side information embedding  $U_s$  through a fully connected network. The general definition of this model is as follows:

$$a_{i=(1,2,3,4)} = \sigma(w_i^h x_1 + w_i^h x_2 + w_i^h x_3 + w_i^h x_4 + b_i^h) \tag{2}$$

$$U_s = \sigma(w_1^o a_1 + w_2^o a_2 + w_3^o a_3 + w_4^o a_4 + b_2^o) \tag{3}$$

$w_i^h$  and  $b_i^h$  represent the weight and bias of the input layer to the hidden layer;  $a_i$  is the output of the hidden layer;  $w^o, b_2^o$  represents the weight and bias of the hidden layer to the output layer;  $\sigma$  represents the Relu activation function which performs better when facing sparse data.

### 3.4. MF Layer

The matrix factorization means decompose user–item interaction matrix into user hidden preference embedding and item hidden feature embedding:

$$V \in R_{n*t} \Rightarrow \{R_u \in R_{n*k} \odot R_i \in R_{t*k}\} \tag{4}$$

$\odot$  means the product of two vectors; the multiplication of the dimensions of two vectors of the same length,  $V$  represents the user–item interaction matrix: it has  $n$  users,  $t$  items;  $R_{ij}$  represents the rating of item  $j$  by user  $i$ . Then  $V$  is decomposed into user hidden preference matrix  $R_u$  and item hidden feature matrix  $R_i$ ;  $k$  represents the feature dimension in here.

### 3.5. The Social Influence Diffusion Layer

In the MF layer, we obtain the implicit preference vector of each user, but the interaction between the user's friends in the social network is not a linear relationship. To measure the dynamic influence among friends in social networks and balance the influence of opinion leaders, we propose a new model of social attention. First, we import each user's hidden opinion preferences into the social impact diffusion layer.

Next for each friend relationship pair, through Equation (5) we can find the weight of their mutual influence:

$$s(i, j) = \frac{\sum_{k=1}^n (X_k \times Y_k)}{\sqrt{\sum_{k=1}^n (X_k)^2} \times \sqrt{\sum_{k=1}^n (Y_k)^2}} \quad (5)$$

$X, Y$  represents the hidden preferences of user  $i$  and user  $j$  in Equation (4).  $n$  means the hidden feature dimensions;  $X_k$  means the value of user  $i$  on the  $K$  dimension.

Lastly, we use SoftMax function to normalize the similarity between users and get the final influence weight:

$$A_{ij} = \frac{e^{s(i,j)}}{\sum_{i=1}^m e^{s(i,m)}} \quad j \in (1, 2, 3, 4...m) \quad (6)$$

$s(i, j)$  represents the similarity calculate in Equation (5), and  $m$  represents the number of friends of the user. Equation (5) calculates the influence weight of user  $i$  on user  $j$ . Lastly we perform normalization and regularization in Equation (6). Through these three steps, we balance the user's trust in a single friend and prevent users from over trusting users, so as to deal with emergency consumption.

### 3.6. FC Layer

For each user  $i$ , the FC layer takes the three embeddings of user  $i$  as input, and outputs a user complete embedding  $u_f$ . We get user interests from different types of input data, to prevent users from falling into the "Information Cocoon Room" (People's information field will be habitually guided by their own interests, to shackle their lives in a cocoon-like Cocoon Room.), reduce the possibility of emergency consumption. We model the FC layer as:

$$u_f = w_1^3 u_s + w_2^3 (u_i + \sum_{j=1}^m A_{ij} \cdot u_j) + b_3 \quad (7)$$

$m$  represents the number of friends of user  $i$ ,  $A_{ij}$  is the normalized weight obtained by Equation (6);  $u_j$  represents the preference of user  $j$ ;  $w_1^3$ ; and  $w_2^3$  represent the weight of the user's side information embedding and the weight of the user's friend influence;  $b_3$  is the bias.

### 3.7. Prediction Layer

We input user features and item features into a fully connected network to make the final prediction. We tested two methods: one is to perform matrix multiplication directly and the other is through a fully connected network and found that the latter has a lower loss.

$$\hat{y} = \sigma(w_1^4 P \odot w_2^4 Q + b_4) \quad (8)$$

$\odot$  means the product of two vectors;  $P \in R_{n,k}$  and  $Q \in R_{t,k}$  represent the user's complete hidden embedding and item's hidden embedding;  $w_1^4$  and  $w_2^4$  represent the weight of the user's embeddings and the weight of the item's embeddings;  $b_4$  is the bias.

### 3.8. A Mechanism to Deal with Misleading Information

We design a classifier to identify the tag identity of the user, if  $y_l = 1$ , we regard it as a misleading user, and if  $y_l = 0$ , we treat it as a normal user. For each user  $u$ , we calculate

the mean square of all rating prediction errors on the interaction item  $M$ . If the error is greater than a certain value, we treat it as a misleading user. and we will update the user representation learned in Equation (7).

$$M_i = \frac{1}{m} \sum_{i=1}^m (y_a - y_p)^2 \quad (9)$$

$$M_i > \frac{1}{n} \sum_{i=1}^n M_i \Rightarrow (y_l^i = 1) \quad (10)$$

$$U'_f = U_f \oplus M_i \quad (11)$$

$m$  represents the number of items that interact with the user;  $n$  represents the number of users. If a user's error is greater than the average error, it will be considered as a misleading user [4]. After the recognition, we can effectively find misleading information in the recommender system, and then update the user representation to prevent the misleading information from affecting more users.

#### 4. Experiment and Results Analysis

In this section, we will show some details of our experiment and our experimental results.

##### 4.1. Datasets

We use four datasets from multiple industries to verify the effectiveness of our model.

LastFM (<https://grouplens.org/datasets/hetrec-2011>), it is a dataset about users listen to music, including social trust information, rating information, and user/item tag information.

MovieLens (<http://movielens.org/>) contains multiple movie ratings by multiple users, as well as movie metadata information and user side information. This dataset is often used in recommender systems and machine learning algorithms. Especially in the field of recommender system, many famous papers are based on it.

Epinions (<http://www.trustlet.org/wiki/Epinions-datasets>), it includes user trust relationship, user rating information, and review information.

Douban (<http://moviedata.csuldw.com>), this dataset is crawled by netizens, which contains movie ratings, user trust relationship, and user side information.

The main statistics of these datasets are summarized in Table 2.

**Table 2.** Statistics of datasets used in this paper.

Data	User	Item	Friends Links	User-Item	Types of Side Information
LastFM	2100	17,635	24,435	92,835	#
Movielens	6040	3883	#	1,000,209	3
Epinions	40,163	139,738	487,183	664,824	#
Douban movie	129,490	58,541	1,692,952	1,683,839	3

##### 4.2. Comparison Methods

To illustrate the effectiveness of our method, we compare the FSCR model with these competitive baselines:

SVD++ [36] assumes that in addition to the explicit historical rating records of users, implicit feedback information such as browsing history and favorite list can also reflect user preferences.

SVD [37] is one of the classic matrix factorization models.

PMF [38] is the probabilistic interpretation version of SVD [37].



TrustSVD [39] is based on SVD++ [36], which further fuses the explicit and implicit influence of trusted users on the predictions of active user items, and achieves good recommendation results.

Social MF [40] is a method of fusing social network into matrix factorization. The user latent vector obtained by Social MF [40] is similar to his neighbor vector, which can realize the spread of trust in the network.

CUNE [41] extracts implicit and reliable social information from easily available user feedback. The extracted information is used to identify each user's Top-K friends for recommendation.

FCR (since some datasets do not have user-trusted information, we simplified our model by removing the social impact diffusion layer).

FSR (for a controlled experiment, the user side information embedding is removed).

#### 4.3. Evaluation Metrics

To show that our model has better performance when facing misleading information, we use two widely used metrics: Root Mean Square Error (RMSE) in Equation (12) and Mean Absolute Error (MAE) in Equation (13).  $y_p$  is the prediction score;  $y_a$  is the true score;  $n$  is the number of users.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_p - y_a)^2}{n}} \quad (12)$$

$$MAE = \frac{\sum_{i=1}^n |y_p - y_a|}{n} \quad (13)$$

#### 4.4. Parameter Setting

Our model is based on Tensorflow 2.0. To obtain objective and real experimental results, we choose 80% of each dataset as the training-set and the remaining 20% as the testing-set.

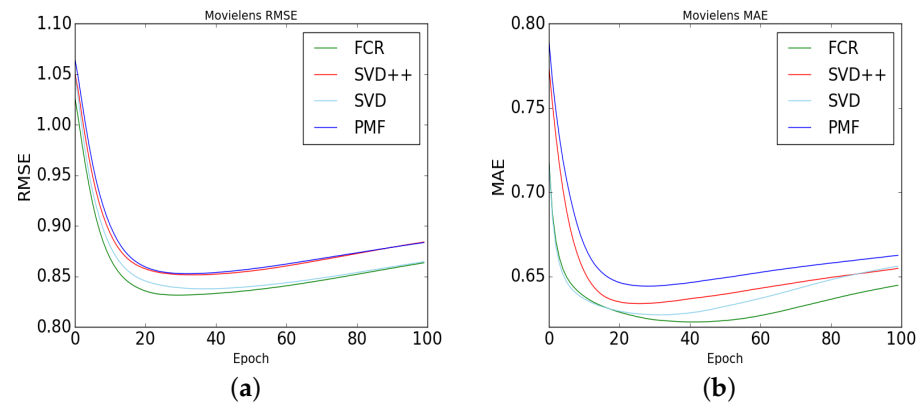
For all the baseline models, we carefully adjust the parameters to ensure optimal performance. The learning rate is 0.01 because if the learning rate is too high, it may overfit, resulting in the error cannot converge; if the learning rate is too small, the training time will be very long. We have tested in advance and found that the best test result is 0.01 between [0.0001, 0.001, 0.01, 0.1]; at the same time, the value of the matrix factorization and the final output size of the embedding matrix are both set to 10. In the experiments, five-fold cross-validation was used for the datasets to get the average performance of the baseline model and FSCR model.

After analysis, we find that the items that interact with users only account for a small part of the total items, the recommender system generally only considers the items that generate interaction, while the items that do not interact are usually discarded. To test our model's resistance to misleading information, we selected 1000 cold-started items in the test-set and let them randomly collide with users. We use this method to simulate the influence of misleading information on the recommendation model and set the attack power to [0%, 20%, 40%, 60%].

#### 4.5. Comparison of FCR with Other Recommendation Models

We compare the performance of FCR and several other recommender baseline models on Movielens datasets. The results were as follows.

In Figure 2, we use Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) to analyze the performance of the proposed model on MovieLens dataset. The results show that modeling the user with the fusion of the user's side information improves the accuracy of the recommender system.

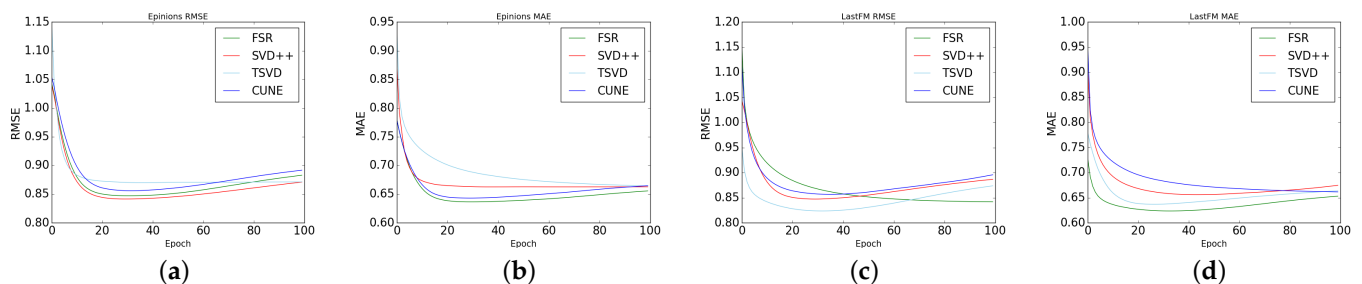


**Figure 2.** Compared with other baseline models, the FCR model (without social influence diffusion layer), the lower RMSE and MAE, the better the performance. (a) Movielens RMSE, (b) Movielens MAE.

#### 4.6. Comparison of FSR with Other Social Recommendation Models

Since there is no user's side information in LastFM and Epinions datasets. To show the effectiveness of our social influence diffusion model, we compare the performance of FSR and several other social recommender baseline models on these datasets. The results were as follows: Epinions: (a,b); LastFM: (c,d).

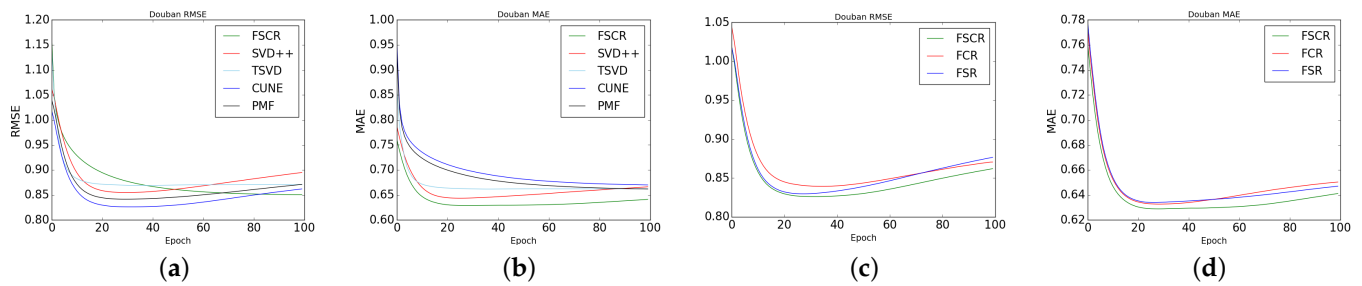
The results in Figure 3 show that our model improves the recommendation results even if we do not learn user preferences from user side information. Compared with the traditional matrix factorization method, our method has a 4–6% improvement on Epinions datasets and LastFM datasets. The results also prove that the recommendation model combining explicit feedback with implicit feedback is superior to the recommendation model considering only explicit feedback.



**Figure 3.** Compared with other baseline models, the FSR model (without introducing side information embedding, but with social influence diffusion layer), the lower the RMSE and MAE, the better the performance. (a) Epinions RMSE, (b) Epinions MAE, (c) LastFM RMSE, (d) LastFM MAE.

#### 4.7. Comparison of FSCR with Other Baseline Models

We will compare the FSCR model with some baseline models. The results are in Figure 4a,b is a comparison of the FSCR model with other baseline models, and (c,d) is a comparison of our three models (FSCR, FCR, FSR).

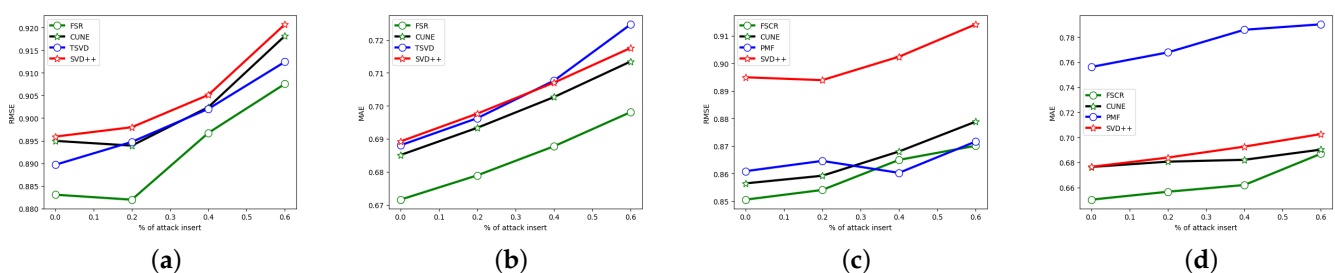


**Figure 4.** Compared with other baseline models, the FSCR model (without introducing side information embedding, but with social influence diffusion layer), the lower the RMSE and MAE, the better the performance. (a) Douban RMSE, (b) Douban MAE, (c) FSCR RMSE, (d) FSCR MAE,

The experiment on Douban movie dataset in Figure 4 proves the effectiveness of our model. First, it can be seen from Figure 4a,b that on Douban data set, the performance of our model is improved by 5–8% compared with other baseline models, which proves that the ability of our model to predict unrated items is worthy of recognition. In addition, in Figure 4c,d, the experimental results show that the performance of the recommender system is greatly improved when user side information is considered. Compared with the model that only considers user–item matrix and trust matrix, the performance of the recommender system is improved by 6–10%.

4.8. The Robustness of Our Model

To show that our model can effectively deal with the problem of emergency consumption caused by misleading information. We adjust the percentage of misleading information inserted. From Figure 5, we can see that our FSCR model is sensitive to misleading ratings. For example, on the Epinions data, MAE in our model only increased by about 9% when the attack intensity is 0.6, while TrustSVD [39] as the baseline model increases by about 12%, and the results are similar in the Douban dataset. This shows that our model has strong robustness against misleading information interference, which is an important feature of the recommender system.



**Figure 5.** The performance of our model is compared with other baseline models under insertion [0–60%] attack. (a) Epinions RMSE, (b) Epinions MAE, (c) Douban RMSE, (d) Douban MAE.

At the same time, the experimental results also show that no matter how hard we try to reduce the impact of misleading information, as the number of misleading information increases, it will eventually affect the performance of the recommender system.

4.9. The Advantages and Disadvantages of FSCR

In this paragraph, we will analyze the advantages and disadvantages of our model. First, we summarize the advantages of our model as follows:

1. Misleading information can only affect users in a short period. In the long run, users’ long-term preferences are not affected.

2. Our model starts from the user's side information, combines the user's consumption preferences, and balances the diffusion of users' friends' preferences in social networks, to avoid users blindly following misleading information.
3. We used a discriminator to identify misleading users in the user-item interaction matrix, and updated their user representation.
4. In this work, the user model comes from a variety of relationships, so it can accurately describe user preferences and make recommendations.

And we summarize the disadvantages of our FSCR model as follows:

1. We only rely on identifying misleading users to deal with misleading information. The misleading information in the recommendation system is diverse, so we need to further optimize our model.
2. The model needs to identify misleading users, which leads to a long training time for the model.
3. Industry recommendation systems often filter out misleading information before training, while FSCR filters out abnormal users and updates their user representation during training.

## 5. Conclusions and Future Work

This paper introduces a new deep recommendation model called FSCR, which can reduce the influence of misleading information on users, avoid users' misleading actions influencing recommendation accuracy to improve the recommender system. We first analyze the problem of misleading information, and find that introducing the misleading information attack directly in the training of the recommender system can improve the robustness, but will affect the recommendation accuracy. Therefore, we propose to combine the users' side information, user trust information and explicit preferences to model the user. The experimental results show that our model has better robustness in the face of misleading information. In the future, we want to consider time series in our models. Because user preferences change over time, this provides a new way for us to model users.

**Author Contributions:** D.Z.; software, validation, resources, writing—original draft preparation, F.Y.; data curation, H.W.; writing—review and editing. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Natural Science Foundation of China under Grant 61801277 and Grant 61373081.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Moscato, V.; Picariello, A.; Sperli, G. An emotional recommender system for music. *IEEE Intell. Syst.* **2020**, 2020.3026000. [[CrossRef](#)]
2. Amato, F.; Castiglione, A.; Mercorio, F.; Mezzanzanica, M.; Moscato, V.; Picariello, A.; Sperli, G. Multimedia story creation on social networks. *Future Gener. Comput. Syst.* **2018**, *86*, 412–420. [[CrossRef](#)]
3. Guo, L.; Jiang, H.; Wang, X.; Liu, F. Learning to Recommend Point-of-Interest with the Weighted Bayesian Personalized Ranking Method in LBSNs. *Information* **2017**, *8*, 20. [[CrossRef](#)]
4. Zhang, S.; Yin, H.; Chen, T.; Hung, Q.V.N.; Huang, Z.; Cui, L. GCN-Based User Representation Learning for Unifying Robust Recommendation and Fraudster Detection. *SIGIR* **2020**, 689–698.
5. Vilakone, P.; Park, D.-S. The Efficiency of a DoParallel Algorithm and an FCA Network Graph Applied to Recommendation System. *Appl. Sci.* **2020**, *10*, 2939. [[CrossRef](#)]
6. Fayyaz, Z.; Ebrahimian, M.; Nawara, D.; Ibrahim, A.; Kashaf, R. Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities. *Appl. Sci.* **2020**, *10*, 7748. [[CrossRef](#)]
7. Liao, X.; Li, X.; Xu, Q.; Wu, H.; Wang, Y. Improving Ant Collaborative Filtering on Sparsity via Dimension Reduction. *Appl. Sci.* **2020**, *10*, 7245. [[CrossRef](#)]
8. Shu, K.; Sliva, A.; Wang, S.; Tang, J.; Liu, H. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explor. Newsl.* **2017**, *19*, 22–36. [[CrossRef](#)]
9. Adomavicius, G.; Tuzhilin, A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 734–749. [[CrossRef](#)]

10. Pazzani, M.J. A Framework for Collaborative, Content-Based and Demographic Filtering. *Artif. Intell. Rev.* **1999**, *13*, 393–408. [[CrossRef](#)]
11. Vozalis, M.G.; Margaritis, K.G. Using SVD and demographic data for the enhancement of generalized Collaborative Filtering. *Inf. Sci.* **2007**, *177*, 3017–3037. [[CrossRef](#)]
12. Li, R.; Wu, X.; Wu, X.; Wang, W. Few-Shot Learning for New User Recommendation in Location-based Social Networks. In Proceedings of the WWW '20: The Web Conference 2020, Taipei, Taiwan, 20–24 April 2020.
13. Liu, Y.; Chen, L.; He, X.; Peng, J.; Zheng, Z.; Tang, J. Modelling high-order social relations for item recommendation. *arXiv* **2020**, arXiv:2003.10149.
14. Sharma, K.; Qian, F.; Jiang, H.; Ruchansky, N.; Zhang, M.; Liu, Y. Combating Fake News: A Survey on Identification and Mitigation Techniques. *ACM Trans. Intell. Syst. Technol.* **2019**, *10*, 1–42. [[CrossRef](#)]
15. Budak, C.; Agrawal, D.; El Abbadi, A. Limiting the spread of misinformation in social networks. In Proceedings of the 20th International Conference on World Wide Web, Hyderabad, India, 28 March–1 April 2011; pp. 665–674.
16. Allcott, H.; Gentzkow, M. Social Media and Fake News in the 2016 Election. *J. Econ. Perspect.* **2017**, *31*, 211–236. [[CrossRef](#)]
17. Takayasu, M.; Sato, K.; Sano, Y.; Yamada, K.; Miura, W.; Takayasu, H. Rumor diffusion and convergence during the 3.11 earthquake: A Twitter case study. *PLoS ONE* **2015**, *10*, e0121443. [[CrossRef](#)] [[PubMed](#)]
18. Gupta, A.; Kumaraguru, P.; Castillo, C.; Meier, P. Tweetcred: Real-time credibility assessment of content on twitter. In *International Conference on Social Informatics*; Springer: Cham, Switzerland, 2014; pp. 228–243.
19. Mejova, Y.; Kalimeri, K. COVID-19 on Facebook Ads: Competing Agendas around a Public Health Crisis. In Proceedings of the COMPASS '20: ACM SIGCAS Conference on Computing and Sustainable Societies, Guayaquil, Ecuador, 15–17 June 2020; ACM: New York, NY, USA, 2020; Volume 285, pp. 22–31.
20. Qian, F.; Gong, C.; Sharma, K.; Liu, Y. Neural user response generator: Fake news detection with collective user intelligence. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence IJCAI-18, Stockholm, Sweden, 13–19 July 2018; pp. 3834–3840. [[CrossRef](#)]
21. Ott, M.; Cardie, C.; Hancock, J.T. Negative deceptive opinion spam. In Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT), Westin Peachtree Plaza Hotel, Atlanta, GA, USA, 9–14 June 2013; pp. 497–501.
22. Ott, M.; Choi, Y.; Cardie, C.; Hancock, J.T. Finding deceptive opinion spam by any stretch of the imagination. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, Portland, OR, USA, 19–24 June 2011; Volume 1, pp. 309–319.
23. Gu, Y.; Shibukawa, T.; Kondo, Y.; Nagao, S.; Kamijo, S. Prediction of Stock Performance Using Deep Neural Networks. *Appl. Sci.* **2020**, *10*, 8142. [[CrossRef](#)]
24. Liu, B.; Zhang, Z.; Yan, J.; Zhang, N.; Zha, H.; Li, G.; Li, Y.; Yu, Q. A Deep Learning Approach with Feature Derivation and Selection for Overdue Repayment Forecasting. *Appl. Sci.* **2020**, *10*, 8491. [[CrossRef](#)]
25. Wang, W.Y. “Liar, Liar Pants on Fire”: A new benchmark dataset for fake news detection. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 422–426.
26. He, X.; He, Z.; Du, X.; Chua, T.S. Adversarial Personalized Ranking for Recommendation. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor, MI, USA, 8–12 July 2018; pp. 355–364.
27. Rendle, S.; Freudenthaler, C.; Gantner, Z.; Schmidt-Thieme, L. BPR: Bayesian Personalized Ranking from Implicit Feedback. In Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, 18–21 June 2009; pp. 452–461.
28. Deldjoo, Y.; Di Noia, T.; Merra, F.A. A survey on Adversarial Recommender Systems: From Attack/Defense strategies to Generative Adversarial Networks. *arXiv* **2020**, arXiv:2005.10322.
29. Tang, J.; Du, X.; He, X.; Yuan, F.; Tian, Q.; Chua, T. Adversarial Training Towards Robust Multimedia Recommender System. *IEEE Trans. Knowl. Data Eng.* **2020**, *32*, 855–867. [[CrossRef](#)]
30. Anelli, V.W.; Deldjoo, Y.; Di Noia, T.; Merra, F.A. Adversarial Learning for Recommendation: Applications for Security and Generative Tasks-Concept to Code. In Proceedings of the 14th ACM Conference on Recommender Systems (RecSys'20), Rio de Janeiro, Brazil, 26 September 2020; pp. 738–741.
31. Di Noia, T.; Malitesta, D.; Merra, F.A. TAaMR: Targeted Adversarial Attack against Multimedia Recommender Systems. In Proceedings of the 2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W), Valencia, Spain, 29 June–2 July 2020.
32. Li, R.; Wu, X.; Wang, W. Adversarial Learning to Compare: Self-Attentive Prospective Customer Recommendation in Location based Social Networks. In Proceedings of the WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston TX, USA, 3–7 February 2020; pp. 349–357.
33. Zhou, F.; Yin, R.; Zhang, K.; Trajcevski, G.; Zhong, T.; Wu, J. Adversarial Point-of-Interest Recommendation. In *World Wide Web Conference*; ACM: New York, NY, USA, 2019; pp. 3462–34618.
34. Kumar, S.; Gupta, M.D. C+GAN: Complementary Fashion Item Recommendation. *arXiv* **2019**, arXiv:1906.05596.

35. Yu, X.; Zhang, X.; Cao, Y.; Xia, M. VAEGAN: A Collaborative Filtering Framework based on Adversarial Variational Autoencoders. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence IJCAI-19, Macao, China, 10–16 August 2019; pp. 4206–4212.
36. Koren, Y. Factor in the neighbors: Scalable and accurate collaborative filtering. *ACM Trans. Knowl. Discov. Data (TKDD)* **2010**, *4*, 1–24. [[CrossRef](#)]
37. Koren, Y.; Bell, R.; Volinsky, C. Matrix factorization techniques for recommender systems. *IEEE Comput. J.* **2009**, *42*, 30–37. [[CrossRef](#)]
38. Salakhutdinov, R.; Mnih, A. Probabilistic matrix factorization. In Proceedings of the NIPS'07: Proceedings of the 20th International Conference on Neural Information Processing Systems, NIPS 2007, Vancouver, BC, Canada, 3–4 December 2007; Curran Associates Inc.: Red Hook, NY, USA, 2007; pp. 1257–1264.
39. Fan, W.; Ma, Y.; Li, Q. Graph neural networks for social recommendation. In *The World Wide Web Conference 2019*; ACM: New York, NY, USA, 2019; pp. 417–426.
40. Hao, M.; Yang, H. Sorec: Social recommendation using probabilistic matrix factorization. In Proceedings of the CIKM '08: Proceedings of the 17th ACM Conference on Information and Knowledge Management, Napa Valley, CA, USA, 26–30 October 2008; ACM: New York, NY, USA, 2008.
41. Zhang, C.; Lu, Y.; Yan, W.; Shah, C. Collaborative user network embedding for social recommender systems. In Proceedings of the 2017 SIAM International Conference on Data Mining, Houston, TX, USA, 27–29 April 2017.