A novel rating style mining method to improve collaborative filtering algorithm

Wei Yang\(^1\), Sheng Hui Guo\(^1\) and Chun Jin Zhang\(^2\)

\(^1\) Key Laboratory for wisdom mine information technology of Shandong Province, Shandong University of Science and Technology, Qingdao, China

\(^2\) Network Information Center (NIC), Shandong University of Science and Technology, Qingdao, Shandong 266590, China

Shenghui Guo (e-mail: 15764225605@163.com).

Abstract. Collaborative filtering (CF) algorithm is widely used in recommendation systems, which makes recommendation based on the neighbors’ interests. Therefore, how to discover the neighbors with similar interests to target user is the core of the CF algorithm. Most existing algorithms discover neighbors by using rating similarity measure, which ignore the differences of users’ rating styles. In this paper, we propose a user rating style mining method and use it to eliminate the rating style differences before calculating a similarity measure. Comparing with the raw similarity measure and another rating style mining method with the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) over amazon movie dataset, we conclude that (i) use our method to eliminate the rating styles differences can improve the prediction accuracy and (ii) our method outperforms other rating style mining method.

1. Introduction

With the recent exponential growth of e-commerce transaction volume and user feedback, discovering useful information from such large-scale data has become increasingly difficult. In order to solve this problem, the recommendation system (RS) has emerged. At present, collaborative filtering (CF) algorithms are commonly employed in RSs, which make recommendations to target user based on his/her neighbors’ interests. Therefore, how to find nearest neighbors is the core of CF algorithms.

At present, the most common existing algorithms discover the neighbors based on rating similarity measures. The most frequently adopted rating similarity measures are Cosine similarity, Pearson similarity, Modified Cosine similarity and so on. In order to estimate neighbors more accurately, many researchers focused on improving these similarity calculation methods.

For example, Srikanth et al. \cite{1} presented a new distance measure by improving Pearson similarity, which can well explain the correlation between users whose ratings are linearly related. Li et al. \cite{2} believed that rating similarity is affected by the number of co-item and the average rating, so he improved Pearson similarity by adopting two factors. Wu et al. \cite{3} estimated the similarity between users suggested with a ratio-based approach. Li et al. \cite{4} integrated the Jaccard coefficients into Cosine similarity and Pearson similarity respectively to get two new similarity measures. Zang et al. \cite{5} considered not only the co-rated items set, but also items rated only by neighbors. Suryakant et al. \cite{6} proposed a CjacMD similarity measure, which combined Cosine, Jaccard, and Mean Measure of Divergence for evaluating sparse datasets.

To address the problem when only a few ratings are available for similarity estimation, Liu et al.
considered both local context information of user ratings and global preference of user behaviour. Further, some researchers introduced the concept such as the co-rated items and the non-common rated items into similarity calculation [8-10]. For example, Wang et al. [8] integrated an asymmetric factor based on the ratio of the co-rated items to all the rated items by each user into similarity calculation. Li et al. [9] introduced another asymmetric factor according to the ratio of co-rated items to non-common rated items by each user. Hu et al. [10] integrated the similarity of items to improve the calculation of users’ similarity in the memory-based collaborative filtering algorithms.

Above literature has proved that these methods can improve the performance. However, most of them ignore that people have different rating styles. For example, there are two users (i.e., users $u$ and $v$) gave their ratings and reviews for an item. Both ratings are 4 stars and the corresponding reviews are "I like it very much. It is really worth the money" and "It is OK, and it is a bit expensive", respectively. From their reviews, we can find that user $u$ likes the item very much and user $v$ considers it is a bit expensive. But both them gave 4 stars. It is indicated that user $u$ has a strict rating style while user $v$ has a relatively loose rating style. However, it is unreasonable to believe that users $u$ and $v$ have similar interest according to their ratings. Because user $u$ will rate less 4 stars if he/her give a review similar to user $v$. Therefore, we must eliminate the differences of rating styles before calculating rating similarity.

This paper proposes a method to eliminate the differences of rating styles and uses it to improve a kind of rating similarity measure with the following key contributions:

1. This paper uses the Bidirectional Encoder Representations from Transformers (BERT) [11] model to mine users’ rating styles and designs a novel method to eliminate the differences of users’ rating styles.
2. This paper uses recalculate a kind of rating similarity measure (i.e., an improved Pearson similarity) after eliminating the differences of users’ rating styles and evaluates the preference with MEA and RMSE over amazon movie dataset.

We develop the rest of this paper as follows. Section 2 introduced some basic knowledge. Section 3 reviews our method. Section 4 describes the experimental details before we conclude this paper in section 5.

2. Basic Knowledge

Before introducing our method, we illustrate some basic knowledge, which will be used in our model. Section 2.1 describes the concept of rating matrix. Section 2.2 introduces the BERT model and section 2.3 illustrates a kind of improved Pearson similarity measure.

2.1. The concept of rating matrix

Definition 1 (Rating matrix) $R_{m \times n} = [r_{ui}]_{m \times n}$ represents the rating matrix that users $User = \{u_1, u_2, u_3, \ldots, u_m\}$ rated items $Item = \{i_1, i_2, i_3, \ldots, i_n\}$, which can be represented as equation (1).

$$R_{m \times n} = \begin{pmatrix} r_{11} & \cdots & r_{1n} \\ \vdots & \ddots & \vdots \\ r_{m1} & \cdots & r_{mn} \end{pmatrix}$$

where, $r_{ui}$ is the rating that user $u$ rated item $i$.

2.2. BERT model

The BERT model [11] is a new language model, which combines the advantages of feature-based (i.e., ELMo) [12] and fine-tuning [13] (i.e., OpenAI GPT) models. The BERT model uses a kind of transformer encoder (figure 1) as the basic structure, which adopts the attention mechanism.

Since the BERT model adopts a multi-layer transformer (TRM) structure (figure 2) to bidirectionally encode word vectors, it can pre-train deep bidirectional representations by jointly
conditioning on both left and right context in all layers. Literature 2 proved that BERT has obtained new state-of-the-art preference on eleven natural language processing tasks. What’s more, Google has already pre-trained the BERT model, we only need to use the given dataset to fine-tune it.

![Transformer encoder](image1.png)  
![BERT model](image2.png)

**Figure 1. Transformer encoder**  
**Figure 2. BERT model**

2.3. An improved Pearson similarity
Literature 2 proposed a kind of improved Pearson similarity considering the affection from the number of co-rated items and users’ average ratings, which is defined as equation (2):

\[ Nsim(u, v) = sim(u, v) \times R(u, v) \times p(u, v) \]  

(2)

where, \( sim(u, v) \) is the traditional Pearson similarity as equation (3), \( R(u, v) \) is the factor to relieve the affection from the number of co-rated items as equation (4), \( p(u, v) \) is the factor to relieve the affection from users’ average ratings as equation (5).

\[ sim(u, v) = \left( \frac{\sum_{i \in I_u} (r_{ui} - \bar{r}_u) \left( \sum_{i \in I_v} (r_{vi} - \bar{r}_v) \right)}{\sqrt{\sum_{i \in I_u} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{i \in I_v} (r_{vi} - \bar{r}_v)^2}} \right) \]  

(3)

\[ R(u, v) = \frac{|I_{u,v}|}{\max(|I_u|, |I_v|)} \]  

(4)

\[ p(u, v) = \left( 1 + \left| \frac{|I_{u,v}|}{|I_u|} \sum_{i \in I_u} |r_{ui} - \bar{r}_u| \right| \right)^{-1} \]  

(5)

where, \( r_{ui} \) represents the rating that user \( u \) rated item \( i \), \( \bar{r}_u \) is the average rating of user \( u \) for all items, \( I_{u,v} \) represents the co-rated item set of users \( u \) and \( v \), \( |I_{u,v}| \) is the size of \( I_{u,v} \), \( I_u \) is the item set that user \( u \) rated, \( |I_u| \) is the size of item set \( I_u \).

3. A novel rating style mining method and its application in improving the similarity measure
Since people have different rating styles, it is unreasonable to calculate rating similarity based on raw rating matrix defined as equation (1). Therefore, this paper proposes a rating style mining method to eliminate the differences in section 3.1. In order to verify the performance of the method, this paper
adopts it into the calculation of a similarity measure reviewed in section 2.3, which will be described in detail in section 3.2. According to the similarity measure, section 3.3 illustrates how to predict the unrated ratings.

3.1. A rating style mining method
According to section 2.2, we know that the BERT model can analyse context effectively. Therefore, we use BERT model to mine users’ rating styles by adding a softmax layer (see figure 3). From figure 3, we can see that the reviews (reviews_u) are taken as input and the corresponding ratings (ratings_u) are taken as output to fine-tune this model. After that, we can get the rating style model of user u (BERT-u). According to the trained rating style model, we can obtain ratings restricted to user u’s rating style based on the reviews by using the transformation function as equation (6):

\[ r_{ui} = B_u(t_{ui}) \] (6)

where, \( r_{ui} \) is the rating that user u rated item i, \( t_{ui} \) is the review corrodong to \( r_{ui} \), \( B_u(\cdot) \) is the rating style transformation function of user u.

According to rating style transformation function of user u and reviews, we can restrict the rating matrix to user u’s rating style as equations (7) and (8).

\[ R_{u_{\text{mon}}} = \begin{bmatrix} r_{u_{11}} & \ldots & r_{u_{1n}} \\ \vdots & \ddots & \vdots \\ r_{u_{m1}} & \ldots & r_{u_{mn}} \end{bmatrix} \] (7)

\[ r_{u_{vi}} = B_u(t_{vi}) \] (8)

where, \( R_{u_{\text{mon}}} \) is the rating matrix in user u’s rating style, \( r_{u_{vi}} \) represents the rating user v will rate item i if his/her rating style is similar to user u, \( t_{vi} \) is the review that user v gave item i.

![Figure 3. The rating style model](image)

3.2. Improving the rating similarity
According to equations 7 and 8, this paper gets the new rating matrixes by restricting the ratings into each user’s rating style, which differ from each other. Therefore, there is not existing rating style differences in the new rating matrixes. Based on the new rating matrixes, this paper recalculates the rating similarity reviewed in section 2.3, which consists of three parts (i.e., traditional Pearson similarity and two factors) defined by equations (3), (4) and (5). Since the calculation of factor \( R(u,v) \) is not related to ratings, we only recalculate the traditional Pearson similarity and the factor \( p(u,v) \),
whose detail steps are as follows. Firstly, according to the new matrixes and equation (3), we recalculate the traditional Pearson similarity as equations (9) and (10).

\[
B \cdot \text{sim}(u, v) = \left( \frac{\sum_{i \in I_u} (r_{ui} - \bar{r}_u) \sum_{j \in I_v} (r_{vj} - \bar{r}_v)}{\sqrt{\sum_{i \in I_u} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{j \in I_v} (r_{vj} - \bar{r}_v)^2}} \right) \sum_{i \in I_u} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v) \tag{9}
\]

\[
\bar{r}_{u-v} = \sum_{i \in I_u} r_{ui} \tag{10}
\]

Where, \( r_{u-v} \) represents the rating user \( v \) will rate item \( I \) if his/her rating style is similar to user \( u \), which is in the new matrix in the user \( u \)'s rating style defined as equation (7), \( \bar{r}_{u-v} \) is the average rating of user \( v \) for all items if his/her rating style is similar to user \( u \).

Then, we recalculate the factor \( p(u, v) \) as equation (11).

\[
B \cdot p(u, v) = \left( 1 + \frac{1}{|I_{uv}|} \sum_{i \in I_{uv}} |r_{ui} - r_{u-v}| \right)^{-1} \tag{11}
\]

Finally, we calculate the new improved Pearson similarity as equation (12):

\[
B \cdot N\text{sim}(u, v) = B \cdot \text{sim}(u, v) \times R(u, v) \times B \cdot p(u, v) \tag{12}
\]

From equation (12), we find that the new similarity is asymmetric, which is consistent with the conclusion drawn in literature 8 and 9.

3.3. Rating prediction
To predict the rating of user \( u \) on unrated item \( i \), we first need to obtain the neighbor set \( N_u \) of user \( u \). According to equation (12), we can compute the similarity values between user \( u \) and other users. Then, according to the user similarity values, the neighbor set is constructed by selecting the first \( N \) users close to user \( u \). Then, the prediction rating is computed according to equation (13).

\[
\hat{r}_{ui} = r_{ui} + \frac{\sum_{v \in N_u} B \cdot N\text{sim}(u, v) \times (r_{vi} - \bar{r}_v)}{\sum_{v \in N_u} B \cdot N\text{sim}(u, v)} \tag{13}
\]

where, user \( v \) is member in the neighbor set of user \( u \).

4. Experiment
The BERT structure used in this paper is BERT-base, which is trained with a free GPU provided by Google. In order to verify whether our model can eliminate the differences of rating styles and whether it outperforms other similar models, this paper selects an improved Pearson similarity (named IPS) reviewed in section 2.3 and CjacMD measure which considers the rating style differences as benchmarks.

4.1. Data preprocessing
This paper implements these algorithms on the Amazon Movies_and_TV_5 data set (http://snap.stanford.edu/data/amazon/productGraph/categoryFiles/reviews_Movies_and_TV_5.json.gz). First, we preprocess this data set by filtering out users with fewer than 100 ratings, then it was randomly divided into 10 portions, 80% of which were taken as training sets, and the remaining 20% were used as test sets. The specific data information is shown in Table 1.
Table 1. dataset information

|                  | Raw dataset | Pre-processed dataset |
|------------------|-------------|-----------------------|
| User             | 124960      | 1443                  |
| Item             | 50052       | 42848                 |
| Reviews          | 1697533     | 367867                |

4.2. Evaluative criteria

This paper selects MAE [14] and RMSE [14] as evaluation criteria, which are defined as equations (14) and (15):

\[
MAE = \frac{\sum_{(u,i) \in \Omega^\text{test}} |\hat{r}_{ui} - r_{ui}|}{|\Omega^\text{test}|} \tag{14}
\]

\[
RMSE = \sqrt{\frac{\sum_{(u,i) \in \Omega^\text{test}} (\hat{r}_{ui} - r_{ui})^2}{|\Omega^\text{test}|}} \tag{15}
\]

where, \(r_{ui}\) is the actual rating in test dataset, \(\hat{r}_{ui}\) is the predicted rating corresponding to \(r_{ui}\), \(\Omega^\text{test}\) is the test dataset, \(|\Omega^\text{test}|\) is the size of the test dataset.

4.3. Experimental analysis

In order to verify whether our model can improve the prediction accuracy by eliminating rating style differences and whether it outperforms other similar models, we implement the BIPs-CF algorithm which is a collaborative filtering algorithm based on a similarity measure improved by this paper, and we select IPs-CF and CjacMD-CF as comparison algorithms, which are the collaborative filtering algorithms based on similarity measures (i.e., IPs and CjacMD). We calculate the MEA and RMSE of all algorithms and draw them into figures 4 and 5. As the number of selected nearest neighbors is well-known to have an important impact on the quality of rating prediction, this experiment compares the prediction performance of these algorithms when the number of neighbors is assigned with the values of 10, 30, 50, 70, 90, 110, 130, 150, 170 and 190.

![Figure 4. MAE among all algorithms with different number of neighbors.](image1)

![Figure 5. RMSE among all algorithms with different number of neighbors.](image2)

Figures 4 and 5 show the results of MAE and RMSE with the horizontal axis representing the number of the neighbors. From those, we can see that the curves of MAE and RMSE first decrease and
then increase with the increasing number of neighbors. Moreover, the MEA and RMSE values of BIPs-CF algorithm are less than ones of IPs-CF and CjacMD-CF algorithms regardless of the number of neighbors. When the number of neighbors is 50, the MAE and RMSE values of all algorithms (i.e., IPs-CF, CjacMD-CF and BIPs-CF algorithms) achieve minimum values (i.e., MAE-0.8796, 0.8345, 0.7739; RMSE-0.8547, 0.8145, 0.7526). At this point, the MEA and RMSE values of BIPs-CF algorithm are least reduced by 12.02% and 11.95% as compared to IPs-CF algorithm and reduced by 7.26% and 7.42% as compared to CjacMD-CF algorithm. It is indicated that eliminating rating style differences with our model can improve the prediction accurate and our model outperforms other rating style mining methods.

5. Conclusion
Collaborative filtering algorithm is widely used in recommendation system, which makes recommendations based on neighbors’ interests. Therefore, how to find neighbors with similar interests to target user is the core of CF algorithms. Since the differences of users’ rating styles will affect the performance of CF algorithm, this paper designs a method to eliminate the differences of users rating styles. In order to illustrate the performance of this method, this paper adopts it into the calculation of a similarity measure. We select the IPs-CF and CjacMD algorithms as benchmarks and compare their performance of prediction with MAE and RMSE. Experimental results show that using our method to eliminate the rating styles differences can improve the prediction accuracy and (ii) our method outperforms other rating style mining method.

Since we only adopt our method into the calculation of a kind of similarity measure, in order to further prove its versatility, we will select other rating similarity measures to improve in the future work. Moreover, the BRET model has powerful performance in natural language processing, so we can use it to analyse the emotional of all topics [15] according to reviews.

Acknowledgments
This paper is supported in part by the Natural Science Foundation of China (No. 71772107, &1403151), Qingdao social science planning project (No. QDSKL 1801138), Shandong Education Quality Improvement Plan for Postgraduate.

References
[1] Srikanth T and Shashi M 2015 A New Similarity Measure for User-Based Collaborative Filtering in Recommender Systems[J] INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY 14.9 6118–28. CrossRef.
[2] Li Rong, Li Mingqi and Guo Wenqiang 2016 Research on Collaborative Filtering Algorithm with Improved Similarity[J] Computer Science 43(12) 213-215+247 Google Scholar.
[3] Wu Xiaokun, Cheng Bo and Chen Junliang 2017 Collaborative Filtering Service Recommendation Based on a Novel Similarity Computation Method[J] IEEE Transactions on Services Computing 10.3 352–365. CrossRef.
[4] Li Wenqiang, Xu Hongji, Ji Mingyang, Xu Zhengzheng and Fang Haiteng 2017 A hierarchy weighting similarity measure to improve user-based collaborative filtering algorithm[C] 2nd IEEE International Conference on Computer and Communications (China: Chengdu) pp 843-846 CrossRef.
[5] Zang Xuefeng, Liu Tianqi, Qiao Shuyu, Gao Wenzhu, Wang Jiatong, Sun Xiaoxin and Zhang Bangzuo Zhang 2017 A New Weighted Similarity Method Based on Neighborhood User Contributions for Collaborative Filtering[C] 1st IEEE International Conference on Data Science in Cyberspace (China: Changsha) pp 376-381 CrossRef.
[6] Suryakant and Mahara T 2016 A new similarity measure based on mean measure of divergence for collaborative filtering in sparse environment[J] Procedia Computer Science 89 450-456 CrossRef.
[7] Liu Haifeng, Hu Zheng, Mian A, Tian Hui and Zhu Xuzhen Zhu 2014 A new user
similarity model to improve the accuracy of collaborative filtering[J] Knowledge-Based Systems 56 156-166 Crossref.

[8] Wang Yong, Deng Jiangzhou, Gao Jerry and Zhang Pu 2017 A hybrid user similarity model for collaborative filtering[J] INFORM SCIENCES 418-419 102-108 Crossref.

[9] Li Qiaoqiao, Lin Zhe and Zhang Fei 2017 Similarity coefficient of collaborative filtering based on contribution of neighbors[C] 2016 International Conference on Machine Learning and Cybernetics (South Korea: Jeju) pp 226-232 CrossRef.

[10] Hu Yan, Shi Weisong, Li Hong and Hu Xiaohui 2017 Mitigating data sparsity using similarity reinforcement-enhanced collaborative filtering[J] ACM Transactions on Internet Technology 17.3 1–20. CrossRef.

[11] Devlin J, Chang M W, Lee K and Kristina T 2018 Bert: Pre-training of deep bidirectional transformers for language understanding Preprint arXiv:1810.04805 Google Scholar.

[12] Peters M E, Neumann M, Iyyer M, Gardner M, Clark C, Lee K and Zettlemoyer L 2018 Deep contextualized word representations Preprint arXiv:1802.05365. Google Scholar.

[13] Radford A, Narasimhan K, Salimans T and Sutskever I 2018 Improving language understanding by generative pre-training Technical report, OpenAI Google Scholar.

[14] Wu Wenmin, Zhao Jianli, Zhang Chunsheng, Meng Fang, Zhang Zeli, Zhang Yang, Sun Qiuxia 2017 Improving performance of tensor-base context-aware recommenders using bias tensor factorization with context feature auto-encoding[J] Knowledge-Based Systems 128 71-77 Crossref.

[15] Dong Luyu, Ji Shujuan, Zhang Chunjin, Zhang Qi, Chiu D.K.W., Qiu Liqing and Li Da 2018 An unsupervised topic-sentiment joint probabilistic model for detecting deceptive reviews[J] Expert Systems with Applications 114 210-223 Crossref.