Noname manuscript No.
(will be inserted by the editor)

Personality-Boosted Matrix Factorization for Recommender Systems

Mazyar Ghezelji · Chitra Dadkhah

Received: date / Accepted: date

Abstract Recommender systems are one of the extensively used Knowledge Discovery in Database techniques which have gained a lot of attention recently. These systems have been applied in many internet-based communities and businesses to make personalized recommendations and eventually gain a higher profit. The core entity in these systems is ratings from users to items but there are many auxiliary pieces of information that can be used to get better performance. The personality of users is one of the most useful information that helps the system to produce more accurate and personalized recommendations. It has been proved that the characteristics of a person can directly affect her behavior. In this paper, We extract the personality of users and propose a mathematical and algorithmic approach to utilize this information. The base model that we use is matrix factorization which is one of the most powerful methods in recommender systems. Experimental results on real-world datasets demonstrate the positive impact of personality information on the matrix factorization technique and also reveals better performance by comparing with the state-of-the-art algorithms.

Keywords Recommender system · Matrix Factorization · Knowledge Discovery · Personality

1 Introduction

The rapid growth of e-commerce and social media has led to a great source of information on the web which causes information overload [1]. This information overload creates some difficulties for the users to find their desired items or information. Eventually, this specific problem will decrease user satisfaction and loyalty [2]. To handle this problem recommender systems have been introduced. A recommender system is a special type of Knowledge Discovery in Databases technique that generates personalized recommendations by analyzing the patterns

Mazyar Ghezelji
Computer Engineering Faculty, K.N.Toosi University of Technology
E-mail: maz.ghezelji@email.kntu.ac.ir

Chitra Dadkhah
Computer Engineering Faculty, K.N.Toosi University of Technology
E-mail: dadkhah@kntu.ac.ir

of users behavior and interests [3,4]. The main idea of the recommendation system is to find a set of items that a user will be interested. Various kinds of recommender systems have been developed by utilizing three main entities in the system which are user, item and user-item interactions [5]. Four type of recommender systems are content-based, collaborative filtering (CF), knowledge-based, and hybrid recommender systems [6]. In the content-based approach, recommendations are produced based on similarities in active user preferences and items attributes. The collaborative filtering approach does not use the attributes features, instead, it calculate the items rating based on the opinions of other people [7]. Knowledge-based recommender system suggests products based on inferences about a user's needs and preferences [6]. Hybrid recommender systems are a combination of different approaches which sometimes gives better results and increases the precision of recommendations. Collaborative filtering has gained many researchers and e-retailers attention due to its attributes: CF does not require domain knowledge, means it can work in systems where gaining attributes of items are difficult or can not be automated [4, 8]. It also can provide a serendipitous recommendation as it does not only recommend items similar to just a specific users history but it takes advantage from a group of users which helps users discover new items [8]. Two major approaches in collaborative filtering are neighborhood-based and model-based, where matrix factorization is the most common model-based approach [6,9]. Neighborhood-based uses several similar users to predict the preferences but matrix factorization represents users and items with a set of latent factors to make it directly comparable. The neighborhood-based approach can find local relations among users and can give a relatively good result when the number of users is not huge and the rating matrix is not so sparse. But, as the size of online communities increases and the number of users grows, the precision of the CF method decreases and it gets harder to implement it at scale. Matrix factorization method can find overall structure unlike the neighborhood-based approach and has good performance in large datasets [4, 10], it also can be integrated with auxiliaries to mitigate cold start problem and improve the precision.

Most of the base recommender systems use ratings from users to items to predict the preferences but there are many auxiliary information like the context of ratings and personality information that could improve this method [9, 11, 12]. Human personality, as the identity of human, affects all aspects of life including behavior in social networks, online shops, etc. The correlation of personality and rating behavior has been widely studied before and the results show that leveraging personality information can help collaborative filtering methods get a better result and solve cold-start problem [8, 11].

The previous methods which use personality information, gain this information usually from other data sources, or by mining the contents of items and profiles of users, but these data sources are not always available. If the only data we have access to is the rating matrix, does it mean that we can't tell the personalities? we answer that it is not a limitation. The personality of users affect their rating behavior and could be obtained from it too. The authors in [13] has shown that personality of a user in terms of optimism and pessimism can be obtained from her ratings for items and other users and it can affect her positive and negative relations with other users. We propose a novel matrix factorization method by utilizing personality information and define the personality of users from their rating behavior. We evaluate our proposed model using Movielens dataset and reveal the advantage of our model through its comparison with other state-of-the-art models.

The remainder of the paper is organized as follows: in section 2, we will review a number of works related to matrix factorization and personality, in section 3, we reviewed the main problem, we also make a hypothesis which is used in matrix factorization, and in section 4 we evaluated and compared our proposed model with other algorithms.

2 Related work

Many researches have shown the success of collaborative filtering recommender systems. Each of the two important types of collaborative filtering, neighborhood-based and model-based, has its positive and negative point, so both of these approaches are being used for either research or industry. Y.Koren demonstrated the importance of explainability in neighborhood-based methods. Koren proposed a neighborhood-based recommender system, which is based on optimizing a global cost function. It maintains the explainability of neighborhood-based method, which is the key to this methods vast usage, and uses implicit ratings which lessens the errors conceivably. He also presented Top-K recommender evaluation technique to distinguish different recommender systems qualities properly [2]. George and Merugu proposed a novel collaborative filtering approach based on weighted Bregman co-clustering algorithm. The main idea of this work is to find neighborhoods faster than matrix factorization approaches, and to be able to use average ratings of co-clusters and users biases in order to generate predictions. the experiments showed that this model can be trained much faster that formal matrix factorization or SVD models [14]. The other side of collaborative filtering, matrix factorization, has developed very fast through recent years. Y.Koren et al. reviewed the theory and applications of matrix factorization method in recommender systems. They boosted the plain matrix factorization method with biases, implicit feedback, and temporal dynamics in order to get better results, which was proved through their implementations [4]. Luo et al. presented a non-negative matrix factorization algorithm for recommender systems. They used a single-element-based approach which results in computational efficiency and ease of use for industrial uses. The result of their algorithm shows that it can outperform classic and also weighted non-negative matrix factorization algorithms in terms of efficiency and accuracy [5]. One of the most important capabilities of matrix factorization is the ability to include different parameters into the learning process with the goal of decreasing error. Hu and Pu addressed cold-start problem by utilizing personality information of users in the matrix factorization collaborative filtering approach. They defined personality as it was introduced in psychology previously; the personality of a person can be defined by five different bipolar dimensions: Openness to Experience, Conscientiousness, Extroversion, Agreeableness, and Neuroticism. Evaluations showed that their proposed cascade method could outperform the classic rating-based collaborative filtering [8]. Personality of users can play an important role in some areas. Khwaja et al. presented an approach for developing an activity recommender system for improving subjective wellbeing. They demonstrated that the link between subjective wellbeing and alignment between activities with personality can considerably improve the accuracy of the recommendation system [15]. Gupta et al. showed the correlations between personality traits of a person (age, gender, lifestyle, etc.) and her musical choice. They believe that personalities can vary from person to person and over time, and also can be extracted and exploited for better recommendations [16]. Khan et al. investigated the interaction of users in social networks like Twitter and IMDB in order to extract the psychological information about them. Their proposed model can use the extracted information for recommending movies. Experiments showed the effectiveness of their model in movie recommendations task compared to other models [17].

3 Our proposed model

Some studies have shown that side information like links between users can be used for increasing matrix factorization accuracy but there are a few studies on finding user attributes

in rating matrix and using them for matrix factorization. To get the maximum result from our limited data source, we defined personality attributes for each user and use them in the optimization process of our proposed model. Our proposed model consists of two parts, *extract personality information* and *Personality-Boosted Matrix Factorization* in terms of optimism and pessimism. First we define the problem , then explain how our proposed method solve it.
Lets assume $U = \{u_1, u_2, \ldots, u_n\}$ is the set of users and $I = \{i_1, i_2, \ldots, i_m\}$ is the set of items, n is the size of users and m is the size of items. We use $G \in R^{n \times m}$ which is the rating matrix. Each entry $G_{u,i}$ of G is a rating from user u for item i. The ratings are in the range of 1 to 5, and 0 means the user has not seen or rated the item.Regarding the data we have, we can define our problem as:
Consider the rating matrix G and personality vector $\vec{P}$, our goal is to fill the entries in G which are 0 by prediction.

3.1 Personality

Optimism and pessimism are two important characteristics of users that have a huge impact on their ratings. Obviously, if a users ratings are mostly more than average she is more optimistic and otherwise [13, 18]. We define the personality of a user as Eq.1.

$$Personality(u) = O_u - P_u. \tag{1}$$

Where O_u is optimism degree of user u, P_u is pessimism degree of user u, and $Personality$ indicates the total propensity of the user to optimism and pessimism which can be a real number in the range of -1 and 1. Personality of 1 means totally optimistic and -1 means totally pessimistic.
The value of O_u and P_u calculate as Eq.2 and Eq.3, respectively.

$$O_u = \frac{|\{r_{ui} | r_{ui} > 3, \bar{r_i} \leq 3\}|}{|\{r_{ui} | r_{ui} \neq 0, \bar{r_i} \leq 3\}|}. \tag{2}$$

Where r_{ui} is the rating that user u have given to item i and $\bar{r_i}$ is the average of all ratings given to item i. This equation indicates that optimism degree of a user depends on the number of high rates that user u had given to less popular items.

$$P_u = \frac{|\{r_{ui} | r_{ui} < 3, \bar{r_i} \geq 3\}|}{|\{r_{ui} | r_{ui} \neq 0, \bar{r_i} \geq 3\}|} \tag{3}$$

Eq.3 indicates that the pessimism degree of a user depends on the number low rates that user u had given to more popular items. Less popular items are items with the average rate of less than number of 3 and more popular item are items with the average rate of greater than number of 3.
In the numerator of both Eq.2 and Eq.3, we don't count $r_{ui} = 3$, because in our model the rating 3 from the user is neutral, neither optimistic nor pessimistic.

3.2 Hypothesis

There are several relations among users characteristics and the decisions that they tend to make. In the case of social networks and recommender systems, the influence of their characters on actions and ratings is undeniable. After analyzing the Movielens 100k user-item matrix, users ratings, and their personalities, we constructed a hypothesis about the relation between personality and ratings. An important point is that this hypothesis is about all users who have rated more items than a threshold. As mentioned before, this dataset has been already pre-processed, so the minimum threshold that we can have is 20. We define T as the set of all users who have rated more items than the threshold. our hypothesis is stated in Eq.4.

$$\forall_{u \in T} P_u > \bar{P_T} \rightarrow \bar{r_u} > \bar{r_T} \tag{4}$$

Where u indicates user, P_u indicates the personality of user u, $\bar{P_T}$ is the average personality of all users in the T set, $\bar{r_u}$ is the average of ratings given by user u and $\bar{r_T}$ is the average of rating given by all users in T set. In other words, our hypothesis is as follows:
among all users in the set T, for each user u if the personality of user u is higher than the average personality of all users in T, then the average rating of user u is higher than the average rating of all users in T.
for testing this hypothesis we use a two-sample t-test on the Movielens 100k dataset [19]. We use two arrays $\vec{tp}$ and $\vec{fp}$ indicating true positive and false positive respectively. In our experiment, we check every user throughout the dataset. If the personality of user u is greater/less than the average personality of all users and average of all ratings by user u is greater/less than the average of all ratings in rating matrix then we add 1 to $\vec{tp}$ and 0 to $\vec{fp}$. Otherwise, we add 0 to $\vec{tp}$ and 1 to $\vec{fp}$. We define null hypothesis H_0 and alternative hypothesis H_1 as Eq.5 according to $\vec{tp}$ and $\vec{fp}$ vectors.

$$H_0 : \bar{tp} \leq \bar{fp} \qquad H_1 : \bar{tp} > \bar{fp} \tag{5}$$

The result of t-test shows that the null hypothesis is rejected with t-statistics equal to 18.5745 and p-value equal to 3.724477401546685e-71. So we can infer that our hypothesis is true, it means if a users personality is more than average, then we can say her average rating is also more than users average.

3.3 Personality-Boosted Matrix Factorization

Matrix factorization (MF) is a very popular model-based collaborative filtering technique. Its scalability, accuracy, ability to integrate regularizations, and ability to provide prediction when there is lack of enough data (cold-start problem), gives us the confidence to use it as the basic model. The main idea of MF is to decompose the rating matrix into two smaller matrixes which the product of them gives back the rating matrix. In this literature, R is the rating matrix with $m \times n$ dimensions, U is the latent user with $m \times k$ dimensions and V is latent items with $n \times k$ dimensions which m is the number of users, n is the number of items and k is the number of latent factors [4]. MF process is as Eq.6.

$$R = U.V^T \tag{6}$$

and therefore finding the U and the V latents is through solving the optimization problem in Eq.7.

$$\min_{U,V} ||W \odot (R - U \cdot V^\top)||_F^2 + \lambda_1 \cdot ||U||_F^2 + \lambda_2 \cdot ||V||_F^2 \tag{7}$$

where W is the weight matrix, usually $W_{ui} = 1$ if there is a rating from user u for item i otherwise $W_{ui} = 0$, $||.||$ is Frobenius norm and $\odot$ is element-wise product of two matrix . The terms $||U||_F^2$ and $||V||_F^2$ are regularization terms which prevent overfitting. The most

well-known method for this optimization problem is stochastic gradient descent which We will discuss it later.
We applied some regularization for the purpose of taking advantage of our hypothesis. Suppose that the personality of a user is more than average personality, but her average rating is less than average rate of all users. In this case we can say the hypothesis is violated. So we add the difference of her average rating and total average rating as a penalty to the optimization function. In another form, the term that we add to the optimization problem to be minimized is stated in Eq.8.

$$\min \sum_{u|P_u>\bar{P}} \max(0, \bar{r} - \bar{r_i})^2 + \sum_{u|\bar{P}>P_u} \max(0, \bar{r_i} - \bar{r})^2 \tag{8}$$

where u indicates every user in the system. The first part of Eq.8is for the case that personality of user is more than average personality and the other part is for the case that personality of user is less than average personality. Putting all the parts together we can write the optimization problem as Eq.9.

$$\begin{aligned}&\min_{U,V} \|W \odot (R - U \cdot V^\top)\|_F^2 + \lambda_1 \cdot \|U\|_F^2 + \lambda_2 \cdot \|V\|_F^2 \\ &+ \sum_{u|P_u>\bar{P}} \max(0, \bar{r} - \bar{r_i})^2 + \sum_{u|\bar{P}>P_u} \max(0, \bar{r_i} - \bar{r})^2.\end{aligned} \tag{9}$$

Because we use max function, There is no closed-form solution for this problem. Therefore we use gradient descent to get a acceptable local minimum. Gradient descent method uses derivative of optimization function in order to shift the solution towards a better one. Using gradient descent requires matrix form equation so we need to rewrite the formulation. First we define new terms for new equation as follows:

- Vector A with m elements where its elements are 1 if $P_i \leq \bar{P}$ and $r_i >= \bar{r}$, -1 if $P_i > \bar{P}$ and $r_i < \bar{r}$ and 0 otherwise.
- Vector X with n elements where its elements are equal to $\frac{1}{n}$.
- Vector Y with m elements where its elements are equal to $\bar{r}$.

Using new terms the formulations are as Eq.10.

$$\begin{aligned}&\min_{U,V} \frac{1}{2} \cdot \|W \odot (R - U \cdot V^\top)\|_F^2 + \frac{\lambda_1}{2} \cdot \|U\|_F^2 + \frac{\lambda_2}{2} \cdot \|V\|_F^2 \\ &+ \frac{\lambda_3}{2} \cdot (A^\top \cdot (U \cdot V^\top \cdot X - Y))^2.\end{aligned} \tag{10}$$

Now we take derivative of optimization function with respect to U and V for updating step in gradient descent process. Showing the optimization function by F, derivative of F with respect to U is stated in Eq.11.

$$\begin{aligned}&\frac{\partial F}{\partial U} = \lambda_1 \cdot U - (W \odot (R - U \cdot V^\top) \odot W) \cdot V \\ &+ \lambda_3 \cdot (X^\top \cdot V \cdot U^\top + (-Y)^\top) \cdot A \cdot A \cdot (X^\top \cdot V)\end{aligned} \tag{11}$$

and derivative of F with respect to V is stated in Eq.12.

$$\begin{aligned}&\frac{\partial F}{\partial V} = \lambda_2 \cdot V - (W^\top \odot (R^\top - V \cdot U^\top) \odot W^\top) \cdot U \\ &+ \lambda_3 \cdot A^\top \cdot (U \cdot V^\top \cdot X - Y) \cdot X \cdot (A^\top \cdot U).\end{aligned} \tag{12}$$

For generalization of the solution we wrote the element-wise product of W with W although it is a matrix of 1s and 0s.
Our proposed method, PBMF (Personality-Boosted MF) is shown as Algorithm 1. After finishing the algorithm, the predicted rating matrix will be $U \cdot V^T$.

Algorithm 1: Our proposed algorithm (PBMF)

Input: the rating matrix **G**, regularization coefficients $\lambda_1, \lambda_2, \lambda_3$, number of latent factors **K**,number of iterations **i**, learning rate α.
Output: **U** and **V**

Calculate W, X, Y ;
Calculate Personalities;
Initialize U, V with random elements;
Define $l = 0$
while l != i **do**
  Calculate A ;
  Calculate $\frac{\partial F}{\partial U}$ regarding calculated parameters ;
  Calculate $\frac{\partial F}{\partial V}$ regarding calculated parameters;
  Update $U \leftarrow U - \alpha \frac{\partial F}{\partial U}$;
  Update $V \leftarrow V - \alpha \frac{\partial F}{\partial V}$;
  Update $l \leftarrow l + 1$;
end
Return U,V ;

4 Evaluations

For evaluation we use Movielens 100k dataset [20]. Movielens is a dataset derived from a non-commercial web-based recommender system with the same name for movies. Each user can rate each movie from 1 to 5. Furthermore users can attach tags based on the content of the movie so that the accuracy of the recommender will be increased.
This dataset consists of 100,000 ratings from 943 users on 1682 movies. The dataset has been pre-processed so each user has rated at least 20 movies. Demographic information like age, gender and occupation is also stated for each user which we don't use in our proposed method.
The information about the personality of users in the dataset are mentioned in Table 1. The procedure that we use for evaluation is cross-validation. To use cross-validation we should first split the dataset into some equal parts then each time set one part as test data and the others as train data.This dataset has been split into 5 test and train parts. Each training part is 80% and each test part is 20% of the whole dataset which are taken randomly and could be used for 5-fold cross-validation. We repeat this procedure for all parts then the final results

Table 1 Statistics of Movielens 100k

	rate	#rates of each user	Personality	Optimism	Pessimism
Min	1	20	-0.79	0	0
Max	5	737	1	1	0.79
Mean	3.52	106.04	0.10	0.23	0.12

are the average result of all tests.

To compare the different methods and demonstrate the error of their predictions we use MAE, MSE and RMSE metrics. MAE is the mean absolute error and is defined as Eq.13.

$$MAE = \frac{\sum_{i \in R} |r_i - \hat{r_i}|}{N} \tag{13}$$

MSE is the mean squared error and is defined as Eq.14

$$MSE = \frac{\sum_{i \in R} (r_i - \hat{r_i})^2}{N} \tag{14}$$

RMSE is the root of MSE as Eq.15.

$$RMSE = \sqrt{\frac{\sum_{i \in R} (r_i - \hat{r_i})^2}{N}} \tag{15}$$

Where r_i is the actual rate, $\hat{r_i}$ is the predicted rate and N is the total number of ratings.

the algorithms that we have selected for comparison are as follows:

- MF [2,4]
  Matrix Factorization is a well-known model-based colaborative filtering algorithm. it is the base for many powerful algorithms because of its flexibility and scalability.
- KNN [2]
  K-Nearest Neighbor algorithm is another model-based colaborative filtering that is based on clustering.
- Co-Clustering [14]
  A dynamic colaborative filtering based approach for recommender systems.
- NMF [5]
  Non-negative Matrix Factorization is a type collaborative filtering based on matrix factorization technique which both users and items are kept positive.

The similarity measure in KNN method is cosine similarity. We set maximum number of k equal to 40 and the minimum number of k equal to 1. The reason for using maximum and minimum for k is that some users have less neighbors than maximum k and further more we can't set users with negative similarity as friends.
Number of factors used in NMF algorithm is 15, regularization terms for users and items are both 0.06 and we have optimized it using stochastic gradient descent for 50 epochs.
For the co-clustering algorithm, the number of user and item clusters are both 3 and the number of iterations optimization loop is 20. As the result of co-clustering and NMF algorithms are dependent on initial state we repeat the experiment with 5 different random initial state and take the average as the final result and for the MF and PBMF the initial state is matrices with all elements of 1.
The iteration of optimization loop in MF and PBMF are both 80, λ_1, and λ_2 in both methods are 0.05. Number of latent factors in MF is 2 and in PBMF is 3. λ_3 in PBMF is 4. we will discuss appropriate λ_1, λ_2, λ_3 and k later in this section.
we divide the algorithms into two groups, 1: matrix factorization based algorithms, 2: the algorithms which are not based on matrix factorization. The detailed comparison of algorithms are shown in table 2.

Table 2 MAE and RMSE of all algorithms

Algorithm	RMSE	MAE
K Nearest Neighbor	0.9801	0.7744
Co-Clustering	0.9665	0.7565
Non-negative Matrix Factorization	0.9669	0.7598
Matrix Factorization	0.9554	0.7581
PBMF	**0.9468**	**0.7493**

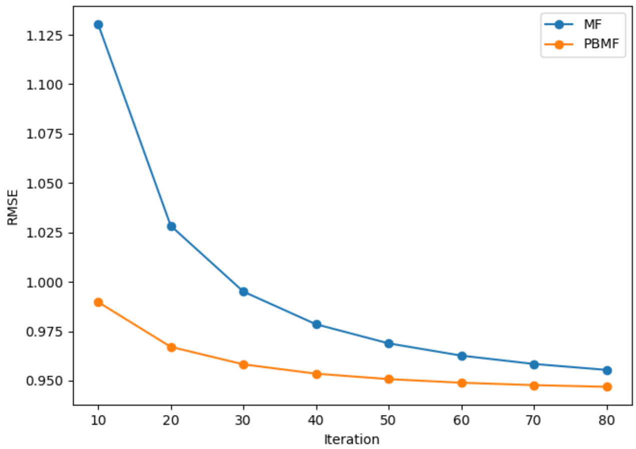

Fig. 1 Comparison of MF and PBMF in different iterations

In order to get the best result using PBMF we should find the optimum regularization terms and number of latent factors. We use the grid search process for tuning the hyperparameters. To do so we use the whole dataset as trainset and calculate MSE for each combination of hyperparameters in a 20 iteration learning loop. The results are shown in figure 2.
We can observe that matrix factorization based algorithms have a relatively better performance compared to other types of collaborative filtering methods. among the non-MF based algorithms, the co-clustering algorithm gives 0.0136 reduction over KNN in terms of RMSE and 0.0179 in terms of MAE.
Among MF based algorithms, PBMF has the best performance with RMSE of 0.9468 which is 0.0086 less than MF and 0.0201 less than NMF. it clearly demonstrates the positive impact of personality information on the performance of MF.
The difference between RMSE and MAE of the co-clustering algorithm is more than other algorithms. from this larger difference it can be infered that the variance of individual errors in the sample is more than others. Comparison of RMSE and MAE for MF and PBMF algorithms through different iterations can be seen from Fig 1. We can see the huge difference between two methods start from the basic steps and it gets less as it converges, so it can be said that for any number of iteration PBMF gives better results than MF.

The hyper-parameter tuning process can be seen from Fig.2. The MSE decreases as the number of factors get higher and then increases by the number of latent factors more than 3. The MSE also decreases as regularization term increases and then increases by the

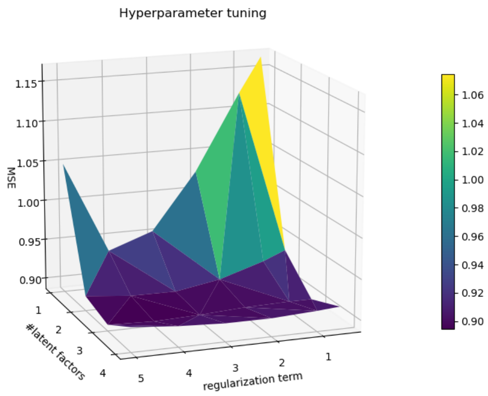

Fig. 2 Hyperparameter tuning

regularization term more than 4. So the optimal regularization term and number of latent factors that we used in our experiments are 4 and 3 respectively.

5 Conclusion

In this paper we proposed a new matrix factorization based personality of users for recommender systems named Personality-Boosted Matrix factorization (PBMF). Experiments on Movielens 100k dataset showed the ability of personality information in terms of optimism and pessimism in empowering the matrix factorization algorithm. The MAE and RMSE of PBMF is smaller than the other algorithm that affected recommendations to be more accurate.
For further studies, we will consider the minimum rates that a user should give and also use personality information in other types of recommender systems.

References

1. Rachana Mehta and Keyur Rana. A review on matrix factorization techniques in recommender systems. In *2017 2nd International Conference on Communication Systems, Computing and IT Applications (CSCITA)*, pages 269–274. IEEE, 2017.
2. Yehuda Koren. Factor in the neighbors: Scalable and accurate collaborative filtering. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 4(1):1–24, 2010.
3. Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Application of dimensionality reduction in recommender system-a case study. Technical report, Minnesota Univ Minneapolis Dept of Computer Science, 2000.
4. Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
5. Xin Luo, Mengchu Zhou, Yunni Xia, and Qingsheng Zhu. An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems. *IEEE Transactions on Industrial Informatics*, 10(2):1273–1284, 2014.

6. Charu C Aggarwal et al. *Recommender systems*, volume 1. Springer, 2016.
7. J Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. Collaborative filtering recommender systems. In *The adaptive web*, pages 291–324. Springer, 2007.
8. Rong Hu and Pearl Pu. Enhancing collaborative filtering systems with personality information. In *Proceedings of the fifth ACM conference on Recommender systems*, pages 197–204, 2011.
9. Mohamed Hussein Abdi, George Onyango Okeyo, and Ronald Waweru Mwangi. Matrix factorization techniques for context-aware collaborative filtering recommender systems: A survey. *Computer and Information Science*, 11(2), 2018.
10. Chen Cheng, Fen Xia, Tong Zhang, Irwin King, and Michael R Lyu. Gradient boosting factorization machines. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 265–272, 2014.
11. Rui Gao, Bibo Hao, Shuotian Bai, Lin Li, Ang Li, and Tingshao Zhu. Improving user profile with personality traits predicted from social media content. In *Proceedings of the 7th ACM conference on recommender systems*, pages 355–358, 2013.
12. He Feng and Xueming Qian. Recommendation via user's personality and social contextual. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 1521–1524, 2013.
13. Ghazaleh Beigi, Jiliang Tang, and Huan Liu. Signed link analysis in social media networks. In *10th International Conference on Web and Social Media, ICWSM 2016*, pages 539–542. AAAI Press, 2016.
14. Thomas George and Srujana Merugu. A scalable collaborative filtering framework based on co-clustering. In *Fifth IEEE International Conference on Data Mining (ICDM'05)*, pages 4–pp. IEEE, 2005.
15. Mohammed Khwaja, Miquel Ferrer, Jesus Omana Iglesias, A Aldo Faisal, and Aleksandar Matic. Aligning daily activities with personality: towards a recommender system for improving wellbeing. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pages 368–372, 2019.
16. Sonali Gupta, Payal Gulati, Surbhi Bhatia, and Rosy Madaan. An automatic approach to music recommendations based on individual personality traits. *Proceedings of the International Conference on Innovative Computing Communications (ICICC) 2020*, 2020.
17. Euna Mehnaz Khan, Md Saddam Hossain Mukta, Mohammed Eunus Ali, and Jalal Mahmud. Predicting users' movie preference and rating behavior from personality and values. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 10(3):1–25, 2020.
18. Ghazaleh Beigi, Suhas Ranganath, and Huan Liu. Signed link prediction with sparse data: The role of personality information. In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 1270–1278, 2019.
19. Jiliang Tang, Charu Aggarwal, and Huan Liu. Recommendations in signed social networks. In *Proceedings of the 25th International Conference on World Wide Web*, pages 31–40, 2016.
20. F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19, 2015.