Global Journal of Engineering Science and Research Management

RESEARCH ON PERSONALIZED RECOMMENDER SYSTEMS BASED ON MATRIX FACTORIZATION

ZHANG SHI-JUN, WANG YONG-HENG*,LIU LI

College of Information Science and Engineering, Hunan University, Changsha 410082, China

E-mail:1187977620@qq.com,wyh@hnu.edu.cn,270818745@qq.com

KEYWORDS: Matrix Factorization, Recommender Systems, Social Network, User Model

ABSTRACT

With the development of social network, there is a large amount of variable information has been made by social network users. We can mine these social data from social network, and find the preference latent relationship between user and items. We would make a model for the user and give a recommended items list to user with a suitable recommender algorithm. That is a variable research subject. So our research would achieve a personification recommender system based on matrix factorization. The research will deal with large-scale user-item ratings matrix. In order to improve the recommender systems' performance we study the social relationship and the implicit feedback of the user. We add a social regularization, demographic information configuration term and users' consumer records as item's latent factor bias terms in the matrix factorization optimization function. Through experiments we recommend more accurate results than CF algorithm and SVD algorithm.

INTRODUCTION

With the rapid development of social network and e-commerce, the research of Recommender Systems becomes more and more popular. Recommender systems appear as a natural language solution tool to overcome the information overload. And they help users discover relevant information in large data sets. Search engine and Recommender Systems plays two different roles. As we all know, search engine is an important tool that people can active search information with key words. Recommender systems as a fundamental tool in on-line services, it recommended a collection of articles presented to the user in order top-N. Recommender systems have achieved much commercial success and becoming increasingly popular in a wide variety of practical applications, for example, in on-line store, such as Amazon and ALiBaBa. In social network, In recent years, more and more users are using social networks, such as Facebook, Twitter and Weibo, In China's most popular social network Weibo, the monthly active users has been more than 167 million, and the context numbers has been more than 100 million. Based on such a huge user and data in social network, Can we build a recommender systems based on social network for information producers and information consumers?

In this paper, we introduce personalized promotion into social network and how to build Recommender systems combined with social relationships between users and explicit feedback. The objective is to achieve high accuracy Recommender systems. Our research focuses on the item recommended and the difficulty is that the data set is large-scale and sparse. Getting user feedback that hidden in vast amounts of data and the user modeling is difficult, great learning of overhead hybrid prediction model parameters.

Recommended studies can be divided into three categories based on micro blogging, (1) information stream Recommended, such as paper [1] the recommended content may be interested in placing forward to help users faster

Foundation item: National Natural Science Foundation of China (61371116)

Author: Zhang Shijun(1991-) Male, master graduate student, The main research: recommendation system, and data mining. WANG Yong-heng, Male, associate professor, doctor, Research direction: large-scale database and data mining, Internet of things complex event processing.LIU Li(1990-) female, master graduate student, The main research: emotion analysis and data mining.

and better access to information in micro blogging. (2) Friends recommended, in paper [2] this area of research is mainly recommended latent friends for the user, expand the user circles. (3) Items recommended. We can make personal recommendation with user's social network information help users and businesses to achieve mutual win. Such as paper [3] the author recommended items to users with LDA [4] model, and in paper [5] the author used data mining methods recommend items to potential users.

USER MODELING

Weibo is the largest social network in China, hundreds of millions of users spread information about themselves on Weibo. We can get an incremental user profile [6] after building User Modeling. I had written a crawling program based on Micro blogging and get the context data and the registration data of users. Before run crawling program we had to preprocessing the user set [7]. Because Micro blogging user presence garbage user (e.g. zombie fans, merchandising account) and agency approval platform (e.g. government agencies, universities, internet news, newspaper media, TV platform, star). These accounts published itself irrelevant information on Weibo, such as zombie fans just as the fan of one account, and agency approval accounts like @HNU that published policies, regulations and notices to its followers. So the object of our research are faced with ordinary users that attention to their account interest, and they published content would associate with user and items, lastly they would interact with their social friends. We could build the user mode when running crawling program after to preprocess the user set.

That preprocessing the user set could reduce the resources consumed by a lot of data processing with worthless users. In our preprocessing system we filtered garbage user according to the user activity and total user behavior. For user authentication (e.g. blue VIP, orange VIP) don't match to research object user and we excluded these from the user set. The crawling program according to the user id could get large amounts of text content and information of the user-related. That is all we want to get the valuable information about user include the history content, comments under the text and the registration information. In our content classification system classified as follows, (a) Users consumed record (b) preference information (c) potential demand information (d) others. Processing text include three steps that is word segmentation, feature extraction and sentiment analysis [8]. For a class of text set we should to extract the user and consumed items recorded. The system does sentiment analysis for b class text set, and there are two main methods that are emotional dictionaries and machine learning methods for short text sentiment analysis. For c class of text set like a class that extract the demanded information between user and items. For d class of text set doesn't need to deal with.

Configure each user's user profile information to the user with the XML document tag storage. User profile allow for these operations (e.g. add, update, delete), and the user profile data include three layers:
1: Rating data make up the initial data of rating matrix M.
2: The demand data of potential users by scoring function transforming into rating data and storing it in matrix M.
3: The data for training prediction model, such as (a) the implicit feedback about users consume record, (b) demographic information, (c) Social relationships.

SCORING FUNCTION

How to process a user profile? That the rating data between user and items in the first layer has stored directly in the rating matrix M. The demand data of potential users by scoring function (1) transforming into rating data and storing it in matrix M, scoring function need to calculate two value set, one is the average value of all items has been rated and the other is the similarity value between users.

$$R_{UI} = \overline{R_U} + Z\sum_{V\in\psi} W(U,V) * (R_{VI} - \overline{R_V}) \qquad (1)$$

In equation (1), $\overline{R_U}$ is an average value of user U has rated for all items. ψ Is a user set that user to rate for the item I, *W (u, v)* is the similarity value between user U and user V in equation (2) with cosine similarity algorithm and in equation (3) with Pearson correlation algorithm [9], Z is a normalization factor Z=$(\sum_{V\in\psi} W(U,V))^{-1}$.When a user to rate someone tend to give high value and someone prefer give low rating, in this situation we should to give a weighted score.

$$\cos(x_u, x_v) = \frac{\sum_{i\in T_{uv}} r_{ui} r_{vi}}{\sqrt{\sum_{i\in T_u} r_{ui}^2 \sum_{j\in T_v} r_{vj}^2}} \qquad (2)$$

$$pearson(x_u, x_v) = \frac{\sum_{i\in T_{uv}} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i\in T_u} (r_{ui} - \bar{r}_u)^2 \sum_{j\in T_v} (r_{vj} - \bar{r}_v)^2}} \qquad (3)$$

Description Calculation algorithm scoring function is as follows, and function floor () represents rounded down, the function ceil () represents rounded up:

ALGORITHM 1: Calculating the rating between the user U and the item I (1-5)

```
Input：user_avg_rating [USER_NUM]: the average value of all items has been rated.
$\psi$ ： A user set that the user had rated for the item I.
$M$ ： With a two-dimensional array to store user and item ratings and we can get the ratings of different users and items from the two-dimensional array.
Output: $R_{UI}$ ： The rating between user U and item I and has been stored in $M$ .
while( $v \in \psi$ )：
    compute K by Eq.3
    K*= ( $R_{VI} - \overline{R}_V$ )
end while
compute $R_{UI}$ by Eq.1
$R_{UI} = \begin{cases} floor(R_{UI}); if\ (floor((R_{UI} - floor(R_{UI}))*10) < 5) \\ ceil(R_{UI}); Otherwise \end{cases}$
```

```
11:   Update: user_avg_rating[USER_NUM]， $M$ .
```

Matrix Factorization

In recent years, matrix factorization algorithm [10] is the most popular algorithms in the field of the recommended systems; especially in the excitation of KDD-CUP and Netflix contest many papers came up with different methods based on matrix factorization model. Matrix factorization takes advantage of a variety of factors such as social relationships and context information, so it can lead to better results recommended and has very good scalability. In simple terms, the matrix factorization algorithm is through dimensionality reduction methods to complement the rating matrix. User ratings behavior can be expressed as a ratings matrix R and R[u] [i] is the rating value of the user U for the item I. In rating matrix many elements that are empty named missing rating. So recommended systems will predict whether the user would rate to some items and how many scores.

4.1 Single value decomposition

We summarize the two most relevant dimensionality reduction algorithms in matrix factorization field: Singular value decomposition (SVD) [11] and principal component analysis (PCA) [12]. SVD in equation (4) is a powerful technique for dimensionality reduction. It is a particular realization of the matrix factorization approach. The key issue in SVD is to find a lower dimensional feature space where the new features represent concepts and the strength of each concept in the context of collection is computable.

$$A_{m\times n} \approx U_{m\times k} W_{k\times k} V^{T}{}_{k\times n} \tag{4}$$

The core of the SVD algorithm lies in the following theorem: Given the m*n matrix data A (m users, n items), we can obtain a m*k matrix U (m users, k concepts), a k*k diagonal matrix W (strength of each concept), and a k*n matrix V (k concepts, n items). The W diagonal matrix contains the singular values, which will always be positive and sorted in decreasing order. The author Yehuda Koren of paper [13] proposed a Latent Factor Model, in equation (5) the rating matrix R was decomposed into two matrices $V^T \in R^{l\times m}, U \in R^{l\times n}$,one of which contains features that describe the user and the other contains features describing the items.

$$\hat{r} = V^T U \tag{5}$$

In equation (6) the Bias-SVD model, μ is the global average score, b_i is the item bias, b_u is the user bias. We through equation (7) train the parameter b_i and b_u.

$$\hat{r} = \mu + b_i + b_u + V^T U \tag{6}$$

$$\min \sum_{(u,i)\in k} (r_{ui} - \mu - b_u - b_i - V_i^T U_u)^2 + \lambda(b_u{}^2 + b_i{}^2 + \| V_i \|^2 + \| U_u \|^2) \tag{7}$$

4.2Hybrid prediction model

In user profile the a class consumption record in third layer as user implicit feedback added into the prediction model. Whether or not the user has rated the item, we use a binary matrix [14] which the value 0 represents the user didn't buy the item and the value 1 represents the user had consumed the item. At this time the hybrid prediction model expression is equation (8) where the R_u represents all items set and y_j is an indicator function.

$$\hat{r}_{ui} = \mu + b_i + b_u + V_i^T (U_u + |R(u)|^{-0.5} \sum_{j\in R(u)} y_j) \quad (8)$$

In user profile the b class demographic information such as age, gender and job. We could predict the preferences between the user and the item with demographic information. In hybrid prediction model we add a double linear model [15] which build model for age and gender in demographic information. Where age can be divided into k groups, each group according to the gender is divided into 2 kinds, so we would get 2k groups. From this definition, g(u) is the user's group number (1<= g(u) <= 2k), $e_{g(u)}$ is a unit vector, *W* is the parameter matrix and β_i is the coefficient vector. At this time the hybrid prediction model expression is equation (9).

$$\hat{r}_{ui} = \mu + b_i + b_u + V_i^T (U_u + |R(u)|^{-0.5} \sum_{j\in R(u)} y_j) + e_{g(u)} W \beta_i \quad (9)$$

In user profile the c class user's social friends list which is the most important context information in recommender systems. The social regularization term makes an assumption that every user's taste is close to the average taste of the user's friends. Where the function sim() in equation (3) is the similarity function to indicate the similarity between user u and user f. F(u) represents the follow friends list of user u, C_{fi} is a indicator function (the value equals 1 if the user f had rated to item i, else the value equals 0). $\bar{r}_f$ is the average rating value and stored in array user_avg_rating[USER_NUM]. Hence, the hybrid prediction model expression is equation (10).

$$\begin{aligned}\hat{r}_{ui} = {} & \mu + b_i + b_u + V_i^T (U_u + |R(u)|^{-0.5} \sum_{j\in R(u)} y_j \\ & + |F(u)|^{-0.5} \sum_{f\in F(u)} sim(u,f) c_{fi} (r_{ui} - \bar{r}_f)) + e_{g(u)} W \beta_i\end{aligned} \quad (10)$$

4.3Parameter learning

In order to learn the model parameters (b_u,b_i,U_u,V_v,W) we minimize the regularization squared error in equation (11). An easy stochastic gradient descent optimization was popularized by Funk and successfully practiced by many others.

$$\begin{aligned}\min \sum_{(u,i)\in K} & (r_{ui} - \hat{r}_{ui})^2 + \lambda (b_u{}^2 + b_i{}^2 + \| e_{g(u)} W \beta_i \|^2 + \| U_u \|^2 + \| V_i \|^2 \\ & + \sum_{j\in R(u)} \| y_j \|^2 + \sum_{f\in F(u)} \| c_{fi} sim(u,f)\ (r_{ui} - \bar{r}_f) \|^2)\end{aligned} \quad (11)$$

Minimization is typically performed by stochastic gradient descent [17], the algorithm loops through all ratings in the training data. For each given rating, the associated prediction error [12] is computed; we modify the parameters by moving in the opposite direction of the gradient, yielding in equation (13) - (17):

$$e_{ui} \overset{def}{=} r_{ui} - \bar{r}_{ui} \quad (12)$$

$$b_u \longleftarrow b_u + r(e_{ui} - \lambda b_u) \quad (13)$$

$$b_i \longleftarrow b_i + r(e_{ui} - \lambda b_i) \quad (14)$$

$$U_u \longleftarrow U_u + r(e_{ui} V_i - \lambda U_u) \quad (15)$$

$$V_i \longleftarrow V_i + r(e_{ui}U_u - \lambda V_i) \tag{16}$$

$$W \longleftarrow W + r(e_{ui} - \lambda W) \tag{17}$$

A general remark is in place. One can get better accuracy by stochastic gradient descent. Several types of implicit feedback can be simultaneously introduced into the model by using extra sets of item factors. For example, we do not expect significant temporal variation for items, which, unlike humans, are static in nature. We would start with a detailed discussion of the temporal effects that are contained within the baseline predictors in the future work.

EXPERIMENT ANALYSES

In this paper, the data source we choose is Weibo. We had written a crawling program based on Weibo and get the context data, the registration data and the follow friends list of users. Finally, we obtain 63641 users, 2484455 content texts. After pre-processing 10847 users were remainder, and building model for every user. Finally, we would obtain the user's user profile. Before to train the prediction model, we get the initialization rating matrix which approximately 10847*1920. The ratings of users were 118920 and the rating matrix's sparsity was 0.5710%.

We use three standard metrics to measure and compare the performance of various recommendation models: Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and Mean Average Precision (MAP@3). The definition as follows:

$$RMSE = \sqrt{\frac{1}{|T|}\sum_{(u,i)\in T}(\hat{r}_{ui} - r_{ui})^2} \qquad MAE = \frac{1}{|T|}\sum_{(u,i)\in T}(\hat{r}_{ui} - r_{ui}) \qquad MAP@3 = \sum_{i=1}^{N} AP@3/N$$

Table 1 shows the evaluating value of varied training data set at prediction model. It can be seen that the bigger scale of training set the smaller the value of RMSE and MAE, which it also represented the accuracy was influenced by the scale of training set. We set the value of K equals 100, K was defined as the latent features number.

Table 1: The evaluating value of varied training data set at prediction model

Train	90	80	70	60	50
RMSE	1.504068	1.512322	1.519119	1.536374	1.549942
MAE	1.281464	1.292177	1.294814	1.307853	1.314353
Density	0.5139%	0.4568%	0.3997%	0.3427%	0.286%

Table 2 shows the evaluating value of varied K value at prediction model. From the table 2 we observe that the corresponding K to the lowest value of RMSE and MAE is a random value. So a suitable value of k would get the most performance value in recommender systems, it also did show the explanation of matrix factorization is more difficult than the other recommender systems algorithms such as based collaborative filtering algorithm. In table2 we set the training set size to 80%.

Table 2: The evaluating value of varied K value at prediction model

k	10	20	30	40	60	80	90	100

RMSE	1.505319	1.509176	1.509736	1.511865	1.513708	1.506729	1.507718	1.512322
MAE	1.283578	1.288589	1.286222	1.290261	1.292298	1.283464	1.287329	1.292177

Table 3 shows the evaluating value of varied algorithms. In table3 we set the same data set and the same training set size (80%). The recommender algorithms contained based user CF, based item CF, SVD and hybrid prediction model. From the table 3 we observe the SVD algorithm is better than CF algorithms in result value of RMSE and MAE, and the hybrid prediction model is the best algorithm in all methods.

Table 3: The evaluating value of varied algorithms

	RMSE	MAE	T(iterations)
Based User CF	1.854921	1.54712	--
Based Item CF	1.872363	1.56071	--
SVD	1.764794	1.41834	100
Hybrid Prediction Model	1.505319	1.283578	100

The other experiment we chose the MAP@3 as the metrics which its value is higher and the performance of model is more accurate. Table 4 shows performances of different multifaceted factorization models. In table 4 we set the same data set and add different bias terms, from the table 4 we observe the performance is increased by add any bias terms, and the social regularization term has the biggest effect for the model.

Table 4: Performances of different multifaceted factorization models

Number	Model description	MAP@3
1	SVD	0.2275
2	Bias-SVD	0.2401
3	2+pairwise ranking train	0.3450
4	3+user implicit feedback	0.3462
5	3+demographic information	0.3471
6	3+social regularization	0.3510
7	6+user implicit feedback	0.3528
8	7+demographic information	0.3541

CONCLUSION

This paper based on Weibo platform, we built the user model for the common user group, through the crawling program we can obtain a large amount of demographic information, based on user contextual information and the valuable data between the user and the item. The hybrid prediction model based on SVD model and combined the above data. In order to improve the recommender systems' performance we study the social relationship and the implicit feedback of the user. We add a social regularization and demographic information configuration terms and users' consumer records as item's latent factor bias term in the matrix factorization optimization function. Through experiments we recommend more accurate results than CF algorithm and SVD algorithm.

ACKNOWLEDGEMENTS

This work was supported in part by National Natural Science Foundation of China (61371116).

REFERENCES

1. Mu Funam. Reserach on recommendation diversity for microblog users[D]. Harbin:Harbin Institute of Technology, 2013.
2. Shi Lili. Research and implementation on Weibo user-oriented content and friends recommendation algorithm[D]. Beijing:Beijing University of posts and Telecommunications, 2014.
3. Zhang Xiaojie. Research on personalized advertising recommendation based on Micro-Blog user interest model[D]. East China Normal University, 2014.
4. Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation[J]. the Journal of machine Learning research, 2003, 3: 993-1022.
5. Wang Li. Design and implementation of microblogging marketing system based on data mining technology[D]. Wuhan:Huazhong university of science and technology, 2013.
6. Manzato M G, Goularte R. A multimedia recommender system based on enriched user profiles[C]//Proceedings of the 27th Annual ACM Symposium on Applied Computing. ACM, 2012: 975-980.
7. CUI Chunsheng. Aggregation analysis of different recommendation input[J]. Acta Analysis Functionalis Applicata, 2014, 16(2): 121-128.
8. Xiong X B, Zhou G, Huang Y Z, et al. Dynamic evolution of collective emotions in social networks: a case study of Sina weibo[J]. Science China Information Sciences, 2013, 56(7): 1-18.
9. Ma H, Zhou D, Liu C, et al. Recommender systems with social regularization[C]//Proceedings of the fourth ACM international conference on Web search and data mining. ACM, 2011: 287-296.
10. Jamali M, Ester M. A matrix factorization technique with trust propagation for recommendation in social networks[C]//Proceedings of the fourth ACM conference on Recommender systems. ACM, 2010: 135-142.
11. Rokach L, Shapira B, Kantor P B. Recommender systems handbook[M]. New York: Springer, 2011.
12. Jolliffe I. Principal component analysis[M]. John Wiley & Sons, Ltd, 2002.
13. Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems[J]. Computer, 2009 (8): 30-37.
14. Yu qiuhong. Relationship recommendation in social networks based on factorizaion machines[D]. Beijing University of posts and Telecommunications, 2013.
15. Tenenbaum J B, Freeman W T. Separating style and content with bilinear models[J]. Neural computation, 2000, 12(6): 1247-1283.
16. Lee W S. Collaborative learning for recommender systems[C]//ICML. 2001, 1: 314-321.
17. Funk S. Netflix update: Try this at home[J]. 2006.