Context-aware Social Popularity based Recommender System

Rajeev Kumar
Student, M. Tech
IET Alwar, Rajasthan, India

B. K. Verma, Ph.D
Associate Prof., CS Deptt.
IET Alwar, Rajasthan, India

Shyam Sunder Rastogi
Student, M. Tech
IET Alwar, Rajasthan, India

ABSTRACT
Contexts and social web information have been recognized to be valuable information for making perfect recommender system. Context-aware recommender systems (CARS) have been implemented in different applications and domains which improve the performance of recommendations. Context-aware approaches have been successfully applied in various domains such as music, movies, mobile recommendations, personalized shopping assistants, conversational and interactional services, social rating services and multimedia. The recommender systems are widely being used for products, content and services recommendations. Successful deployment of recommender system in social web and many commercial website like Amazon.com, flipkart, HomeShop18 and numerous different sectors have already done. The growth of the social web has revolutionized the architecture of sharing and association in the web, making it essential to reiterate recommendation. If recommender systems have established their key role in providing the user access to resources on the web, when sharing resources has turn into social, it is likely for recommendation techniques in the social web should consider social popularity factor and the relationships among users to compute their predictions. In this paper contextual information are being included in social popularity based SVD++ model to improve accuracy and scalability of recommendations.

Keywords
Contextual information; SVD; Social Popularity

1. INTRODUCTION
Recommender systems (RS) recommend things to users depending on preferences generally expressed in the form of numeric ratings of like-minded people. It has become a very important tool in many commercial and e-commerce websites like Amazon, Netflix, facebook, Twitter, LinkedIn, and Epinions. Most recommender systems depend on collaborative filtering methods [2, 25], that predict a user's preference from an item by extracting the patterns from the past rating information of other similar users and/or items. While collaborative filtering has turn to the standard method for the recommendation problem, conventional recommender systems only use ratings of related users/items to extract recommendations without considering any other information. It recommends things to users and do not concern about the situation and contextual information when recommendations take place. For example, when an online travel agency suggest a vacation package, it is essential to know when the person makes a plan to go on vacation. In recent times, some business organization started considering the contextual information. Context-Aware Recommender Systems (CARS) deal with contextual data (e.g. time, place, social relation, and mood) related to the collected preferences. In this way, CARS can differentiate the interest a user may have in a particular thing within different contexts and situations. There are several approaches which deal with contextual data. These are pre-filtering, which considers that the contextual information is used to filter out unrelated ratings before they are applied for making recommendations with typical methods; those based on contextual post-filtering, considers that the contextual information is used after the typical non-contextual recommendation methods are applied to the recommendation data.; and those based on contextual modeling, which include contextual data into the model used for generating recommendations. It can be seen from previous research that pre-filtering, post-filtering and contextual modeling have been very successful and they are able to improve the result of recommendations in different areas and applications.

In academia, several studies demonstrated that context information made a lot of changes in a consumer buying activity. Many researches on consumer modeling proposes that incorporating context in a consumer activity model can bring improvement to predict her activities in several cases since it permits the recognition of more identical patterns in the data related to the buying history of a customer [2]. Therefore, accuracy of predicting consumer interests should depend on the amount of relevant contextual information [24]. This led us to directly use famous Machine Learning algorithms for contextual modeling, and evaluate pre-/post-filtering with context modeling. The main problem is that how to evaluate CARS is the unavailability of contextual datasets. Collecting contextual information require an additional effort from the user to explicitly describe the up to date context, or system/machine requirements to automatically extract up to date context, e.g. time and location, or by exploring the user's relations with the system. This fact makes it hard to access to contextual information which are truly important for evaluation.

As social popularity of items or things is affected by various factor that need to consider. First things is that if some items or things are popular for particular location and being liked many people of that region then it must need to include location information. For example, X enters "*Which is the nearest shopping mall*?" in Google search engine then it must use the location information in which the user are present and search result are displayed accordingly. Similar concept should be used in recommendation system. Similarly if some item or things are being liked in some particular time then it must be incorporated in recommender system to improve the recommendation result. Suppose, Y want to search "*Which is the best time for picnic*?" then search result should be different for different time because if some particular time are best suited for some location then it must not be favorable for other. That why time information are very important factor for recommender system.

The paper is arranged in following manner. The next section talks about related work. Section 3 discusses about proposed work. Section 4 explains excremental evaluation method and Section 5 presents results and discussion. In the last section, conclusion and future work are specified.

2. RELATED WORK

Collaborative filtering (CF) [1] and content based filtering have been commonly used to help users discover the most valuable information. With development of social webs, researchers propose trust-based and influence-based methods which consider the user relationship for recommendations. Contextual information about emotions is researched by Gonzalez et al [8] in recommender systems in 2007. They specified that sentiments are important for user's judgment making in recommendation process. The users always express their decisions together with sentiments. With the rapid development of context-aware recommender systems, sentiment turns out to be one of essential and accepted contexts in different kinds of fields, especially in the music and movie fields. For music recommendation, the sentimental context is interesting because it can be used to set up a bridge between music and things from other different fields, and make cross-media recommendations. Movie recommendation is another field where sentiment becomes popular in recent years.

In 2010, Yue et al [16] described that how mood related movie similarity using matrix factorization approach can be used for context aware movie recommendation. More recent research [18, 13] motivates the leaning on sentiments as contexts to help contextual recommendations. Research has verified that sentiments can be dominant contextual factors in making recommendations, but none of them explain that how sentiments work together with recommendation algorithm the usage of sentiment factor in the recommendation process. Contextual information has informed to be valuable for providing more correct prediction in various application areas including recommendations. Contexts can be collected in various ways, such as by explicitly extracting relevant contexts from users/items, by implicitly get from data or location, or by deriving using statistical techniques, or data mining/machine learning,

Adomavicius et al. [9] offered a multidimensional recommender system based on multiple dimensions, i.e., user/item dimension as well as different contextual information. Contextual information is preprocessed, firstly before being used, by using various statistical methods such that only the contexts that are truly useful are selected for recommendation. Recent works have concentrated on making models that directly incorporate contextual information with conventional user-item-rating relations. For example, Karatzoglou et al. [19] suggested a multiverse recommender system by modeling the data as a user-item-context N-dimensional tensor. Several matrix factorization techniques have been suggested for collaborative filtering approach. The matrix factorization techniques are based on the concept of user-item matrix with low dimensional latent factors. Determine such types of factor are very important because they affect social web relationship, as well as influence behavior. Knowing that influence is a subtle force that governs the dynamics of social webs, influence-based recommendation involves interpersonal influence provided by the sender and receiver into social recommender system.

It is proved in much research that Tensor Factorization (TF) is the most accurate model-based CARS technique. TF extends the traditional 2D matrix factorization model into an nD matrix of the same model, which is known as tensor factorization. The multi-dimensional matrix [23] is factored into lower-dimensional version, where the user, the item and each contextual element are shown with a lower dimensional feature vector. Tucker decomposition is used to factorize the tensor. However, this model is only appropriate for definite contextual information. An additional improvement was suggested to provide to all types of contexts. However, even though the authors state that the suggested model is able to minimize computational complexity for huge user-item rating matrix, the model may still endure from scalability problem. One possible solution to improve scalability problem is to divide the original matrix before applying any factorization techniques.

Zhong et al. [3] suggest a contextual collaborative filtering approach, which is also known as RPMF, to allow context-aware recommender system. The theory behind this model is that contextual information is determined by the user-specific and item-specific latent factors. On the basis of it, a random partition is used to divide the user-item rating matrix by classifying users and items with like contexts, and then matrix factorization are applied to generate sub-matrices. Liu et at. [2] Suggest a SoCo method which explicitly makes use of context for processing and works on only the leaf node and also use the social information.

Ma et al. [12] suggest a matrix factorization technique with social regularization. But this work only consider user aspect vectors from social side but do not take into account users' individual side, which create the technique lack of contextual information to further improve the recommendation accuracy. Xu et al. [5] suggested to group users and items such that similar type of users and their items are grouped. Contextual information is then used by applying collaborative filtering (CF) to improve top-N recommendation value.

Yang et al. [6] first presented that a user may trust different subset of friends regarding different areas, and then proposed a group specific circle-based model to construct context-aware recommendation. However, these works consider very basic contextual information (e.g., category/group). Akther et al. [4] planned architecture to gather contexts and social web information for personalized recommender systems. The authors thought that how important data is extracted and stored but overlooked how such data is efficiently united from an algorithmic point of view. Jiang et al. [7] proposed to combine social contexts (individual liking and social influence) into a matrix factorization model. However, such contextual information is only associated to social relationships, so non-social contexts are mostly ignored

3. SOCIAL POPULARITY WITH CONTEXTUAL INFORMATION

In this section, social popularity aided with context information based recommender systems is presented. Preliminaries notations are presented in Section 3.1 then matrix factorization technique using SVD model are presented in Section 3.2. Our context information based social popularity based recommendation approach and its social webs based enhancement is presented in Section 3.3 and Section 3.4 respectively.

3.1 Preliminaries

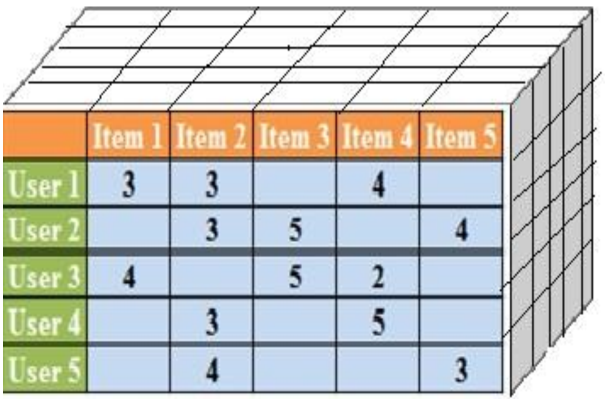
Traditional recommender systems use matrix factorization approach to make recommendation. There are various well known matrix factorization algorithm are available like Tensor, Non-negative matrix factorization, Online-NMF, singular value decomposition (SVD) [13]. They all usually based on the user-item-rating matrix decomposition for recommendations. However, due to availability of contextual information in various systems, it provides a new way to improve recommendation. Contextual information can be broadly classified into two parts. First one is static context, that describes characteristics of a user, e.g., gender, age, Religion, job, etc. or an item, e.g., class, rate, physical properties, etc.; and second one is dynamic context [22] that describes changing information that is related with a rating (e.g., a user's mood or place where he/she rates an item). Social webs have made people to get opinion or reviews of different products that affect user preference for an item. People want to share their own views as well as they want to find similar taste from his/her friends. Therefore, we integrate contextual information and social popularity based information into a SVD based factorization model to improve recommendations quality.

Let us suppose that user can be defined as user set by $\mathrm{U} = \{\mathrm{u}_1, \mathrm{u}_2, \mathrm{u}_3, \dots \dots . \mathrm{u}_\mathrm{m}\}$ and the item can be defined as item set by $\mathrm{V} = \{\mathrm{v}_1, \mathrm{v}_2, \mathrm{v}_3, \dots \dots . \mathrm{v}_\mathrm{m}\}$. There are no restriction that which user will rate which item. It is assume that value of rating will be a discrete value on the scale of 5 or 10. So rating can be defined as a function of user, item as well as context.

$$r{:}\, user \,*\, item \,*\, context \rightarrow rating \tag{1}$$

	Item 1	Item 2	Item 3	Item 4	Item 5
User 1	3	3		4	
User 2		3	5		4
User 3	4		5	2	
User 4		3		5	
User 5		4			3

(a) User-item-rating Matrix



(b) Contextual Information in Z-axis

Figure 1: Context-based social recommendation.

A rating provided by user u on item i can be defined as $\mathrm{r}_{\mathrm{u,i}}$ and all rating $\mathrm{r} = \{\mathrm{r}_{\mathrm{u,i}} \,|\mathrm{u}_\mathrm{m} \in \mathrm{u}, \mathrm{i}_\mathrm{n} \in \mathrm{i}\}$ makes a user-item-rating matrix. It is also assume that set of contextual information related with each rating $\mathrm{r}_{\mathrm{u,i}}$ denoted *by* $c_i = \{c_1, c_2, c_3, \dots \dots\}$. *A*ll ratings have the identical contextual information vector and no constraint on the value domain of each type of contextual information, i.e., both discrete values and continuous values are suitable. Social webs can be defined a directed graph $G = (N, E)$, where edge set E signifies the relations between users (u). the friend set of a user u_m by $F_u \subset u$.

3.2 Singular Value Decomposition (SVD)

Fundamentally, the aim of matrix factorization is to factorize a matrix into two (or more) matrices and later they can be converted into the original matrix by multiplying the factorized matrices. In the case of recommendation problem, a matrix factorization model factorizes a user-item-rating matrix $R \,\varepsilon\, R^{m \,x\, n}$ where m is the number of users and n is the number of items, into user matrix $U \in R^{d \,x\, m}$ and item matrix $V \in R^{d \,x\, n}$:

$$R \approx U^T\, V \tag{2}$$

Where d is the length of a latent factor vector which marks a user or an item. For a user u, the elements of U (i.e.,U_u) determine u's concern in items that have higher values on the subsequent latent factors; for an item V, the elements of V (i.e.,V_v) determine the strength of association [14] between V and the subsequent latent factors. Therefore, the resultant $U_u^T\, V_v$ represents the correlation between user u and item v, i.e., u's preference for v's, considering all latent factors.

In order to estimate R, the following function is defined, considering sparseness of the user item rating matrix because lot of user do not rate different items so huge amount of rating values R are missing.

$$\arg\min_{\mathrm{U,V}} \sum_{i=1}^{m} \sum_{j=1}^{n} X_{i,j}\, \left(R_{i,j} - U_i^T\, V_j\right)^2 \tag{3}$$

Where $X_{i,j}$ is 1 if user i has rated item j and 0 otherwise. Moreover, a regularization term is added with the equation (3) to minimize overfitting.

$$\arg\min_{\mathrm{U,V}} \sum_{i=1}^{m} \sum_{j=1}^{n} X_{i,j}\, \left(R_{i,j} - U_i^T\, V_j\right)^2 + \lambda(||U||_F^2 + ||V||_F^2) \tag{4}$$

Where $||A||_F^2$ is the Frobenius norm of g * h matrix, calculated by $\sqrt{\sum_g^X \sum_h^Y \left||A_{g,h}|\right|^2}$. The regularization can be control by parameter λ.

Above Eq. 3 can be solved (i.e., minimized) using two techniques: (i) stochastic gradient descent (SGD) that recursively updates user-related latent factors and item-related latent factors and (ii) alternating least squares (ALS), which joins U (or V) and optimizes V (or U), and then rotates, iteratively.

3.3 Contextual information based recommendation

Firstly, we have to understand that how to integrate contextual information to get better recommendation quality without considering social relations. Random decision trees algorithm are applied to combine different contextual information, which is one of the most perfect learning algorithms to make multiple decision trees arbitrarily. The motivation behind this method is to divide the original rating set R (i.e., user-item-rating matrix) such that ratings with the identical contexts are grouped. Since generated in identical contexts, ratings in the similar cluster are likely to be better associated among each other than those in original rating matrix (i.e., the missing ratings can be derived more accurately).

The partition process [15] keeps on until one of following conditions is met: (i) all contextual information is iterated; (ii) the limitation on the height of a tree has been achieved; (iii) there are not an enough number of ratings to divide at the current node. After dividing, the ratings R are sorted based on different contextual information. Note that in various decision trees, the training ratings are categories differently, given that contexts (and their values) are selected randomly at each level of a tree. When predicting a missing rating $\bar{R}$, it is assumed that there are t decision trees, and in each decision tree, $\bar{R}$ is categorized to the rating subset (i.e., user-item-rating sub-matrix) $\bar{R}_i \subset R$ according to R contextual information. For each $\bar{R}_i$, it is further decomposed and finds the factorized user and item matrices $\bar{U}_i$ *and* $\bar{V}_i$ that can be used to predict the missing rating$\bar{R}_m$.

$$\ell_1 = \arg\min_{\bar{U}_i,\bar{V}_i} \sum_{\mathrm{j}=1}^{\bar{U}_i} \sum_{\mathrm{k}=1}^{\bar{V}_i} X_{j,k}\left(\bar{R}_{i,j,k} - \left(\bar{U}_{i,j}\right)^2 \bar{V}_{j,k}\right)^2 + \lambda(||\bar{U}_i||_F^2 + ||\bar{V}_i||_F^2) \qquad (5)$$

$$R_{m,i} = (\bar{U}_i)^T \bar{V}_i \qquad (6)$$

Finally, the predictions from t decision trees [16] are merged to make the final prediction for $\bar{R}_m$.

$$\bar{R}_m = \frac{\sum_{i=1}^{t} R_{m,i}}{t} \qquad (7)$$

3.4 Social popularity integration with contextual information

From the previous sections, which described the context based recommendation model, an improved version of previous model have been described in this section by considering social popularity [17] to improve recommendation quality. As in real world, when someone decides whether or not to purchase product e.g. pen drive, mobile, moves ticket , he/she often ask for suggestion from his/her friends, whose tastes are likely to be similar with him/hers. By getting views from multiple friends, one can able to take better decisions. Although friends' opinions give valuable information to help in making high quality recommendation for users, most existing works either employ/combine all available social popularity information without information filtering [18] or do not examine how to exactly measure taste similarity between two users. In order to deal with these issues, following the approach suggested, which establish a social popularity to restrict taste difference between a user and his/her friends. In the real world, a user could have hundreds or even thousands of friends; it is thus worthless to deal all friends (and their information) equally because some friends can have similar tastes with the user, while some others may have completely different tastes. In order to deal with such social taste heterogeneity [18], the following social popularity term considers taste similarity between a user and each of his/her friends:

$$\alpha \sum_{j=1} \sum_{f \varepsilon j} s_c(j,f) \left\| U_{i,j} - U_{i,f} \right\|_F^2 \qquad (8)$$

where α is a constant controlling the degree of social popularity, $s_c(j,f)$ shows the taste similarity between user u_j and one of his/her friends u_f based on their past rating history as well as contextual information. The higher value of similarity score shows the higher correlations between users u_j and u_f, while lower similarity score means that both users are very dissimilar to each other. Similarity score can be calculated using different method but here log likely hood based similarity [18] are used to find out the correlation score. Detail descriptions of log likely hood are not given there. It is better that similarity score are normalize from the range [-1, 1] to [0, 1] before applying it to the recommendation model. From equation (8), social popularity terms have been integrated into equation (5) to consider these terms for better recommendations.

$$\ell_2 = \arg\min_{\bar{U}_i,\bar{V}_i} \sum_{\mathrm{j}=1}^{\bar{U}_i} \sum_{\mathrm{k}=1}^{\bar{V}_i} X_{j,k}\left(\bar{R}_{i,j,k} - \left(\bar{U}_{i,j}\right)^2 \bar{V}_{j,k}\right)^2 + \alpha \sum_{j=1} \sum_{f \varepsilon j} s_c(j,f) \left\| U_{i,j} - U_{i,f} \right\|_F^2 + \lambda(||\bar{U}_i||_F^2 + ||\bar{V}_i||_F^2) \qquad (9)$$

Eq. (9) can be solved by executing gradient descent in $\bar{U}_{i,j}$ *and* $\bar{V}_{j,k}$, which is iteratively updated.

$$\bar{U}_{i,j} \leftarrow \bar{U}_{i,j} + \gamma \frac{\partial \ell_2}{\bar{U}_{i,j}} \qquad (10)$$

$$\bar{V}_{j,k} \leftarrow \bar{V}_{j,k} + \gamma \frac{\partial \ell_2}{\bar{V}_{j,k}} \qquad (11)$$

Where λ is the learning rate.

4. EXPERIMENTAL EVALUATIONS

This section examines the proposed model applying different dataset as well as applying different evaluation matrices. Here, dataset are specified firstly then evaluation matrices are presented. Then results are shown with the help of different graphs and discussed the results. Finally conclusions are presented with future work.

4.1 Dataset

Douban is one of the leading Chinese social web for sharing reviews and suggestion for books, films and music. Each client or user has to offer ratings which are on the scale of one star to five stars, to the books, films, music, representing his/her interest on the things. A timestamp is linked with a rating. A user, although has not used an item (i.e., no rating is needed), may still convey his/her concern by providing "wish" (e.g., wish to see the movies). A social web is rendered reviews which are important and valuable for those users who follow another user. Table 1 shows the statistics of the Douban dataset. Note that we make use of explicit ratings, i.e., the "wish" terms are not considered to be ratings.

Table 1: Statistics of the Douban dataset

	No. of ratings	No. of users	No. of items
Book	812,037	8,598	169,982
Movie	1,336,484	5,227	48,381
Music	1,387,216	23,822	185,574
All	3,535,737	25,560	403,937

Douban dataset are selected because it include not only timestamp related and other contextual information, but also social popularity information, thus is appropriate for evaluating the performance of our proposed model, which make use of various types of information. There are also another publically available dataset like MovieLens which also include timestamp information [19]. But social popularity information is not available over here. So MovieLens dataset of 1M have been selected for comparing the performance of proposed model. The dataset comprise of about 1 million ratings of about 3952 movies made by 6040 users. Ratings are also provided on a 5-star scale, and each user has provided at least 20 ratings. For both datasets, 85% of ratings data has been selected to train recommendation model and evaluate their performance using the rest 15% of the ratings.

4.2 Evaluation Metrics

There are various types of measures are available to evaluate the performance of the recommender system. The quality of prediction is main factor which affect the result of recommendation. Here, two important metrics are used to compute and evaluate the performance of proposed recommendation techniques. Mean Absolute Error (MAE) [20] are defined as Eq. 12 which are given below.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|r_{u,i} - \bar{r}_{u,i}\right| \qquad (12)$$

Root Mean Square Error (RMSE) is defined using Eq. 13 that is given below.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(r_{u,i} - \bar{r}_{u,i})^2} \qquad (13)$$

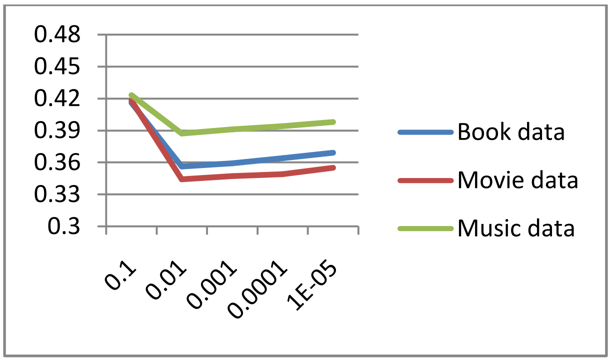
Where N is the total number of predictions, $r_{u,i}$ is the actual rating of an item i given by user u and $\bar{r}_{u,i}$ is the subsequent predicted rating.

The lower value of MSE and RMSE [21] signify that recommender system has been performing more accurately than the higher value of these. Finally, recommender system will help to gain revenue from business by attracting more users to use it.

5. RESULTS AND DISCUSSION

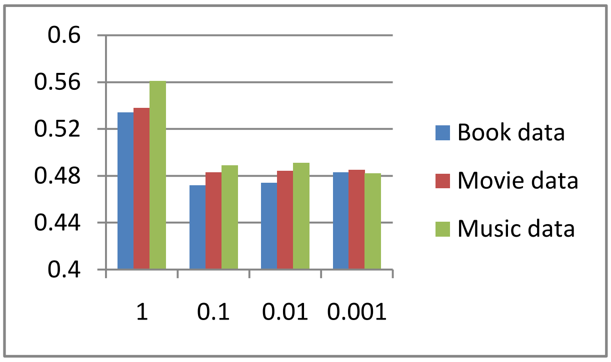
5.1 Performance on Douban dataset

To show the impact of various parameters, firstly Douban dataset have been evaluated and regularization constant has been set $\lambda = 0.1$ by cross validation. Table 1 show that when different subset of the dataset (i.e., book data, movie data and music data) is provided that how the performance of novel model varies with different values of parameter α, which tell how much social popularity information is included into novel model. Here latent factor vector dimensionality and the number of iterations for solving a matrix factorization model are set to 5 and 10 respectively.



(a) MAE

From experiment, it is observe that when α increases, both MAE and RMSE first decrease, and then turn into somewhat stable (but slightly increase) when α reaches a certain threshold, i.e., about 0.01. It is concluded that social popularity information is able to efficiently improve recommendation quality and α = 0.01 is a proper threshold that well balances the user-item-rating matrix and social popularity information.



(b) RMSE

Fig 1: Impact of parameter α (λ = 0.1, dimensionality = 8, iteration # = 10)

5.2 Performance on MovieLens-1M dataset

Test conducted on the Douban dataset show that combining contextual information and social popularity information really improves recommendation quality. However, in some application, no social popularity information is provided (e.g., Netflix). In order to calculate the performance of novel model when only traditional information is available, another set of tests are conducted on MovieLens-1M dataset. All matrix factorization based models perform better than traditional memory based algorithms, which again show the advantage of the latent factor models with lower RMSE value at different time.

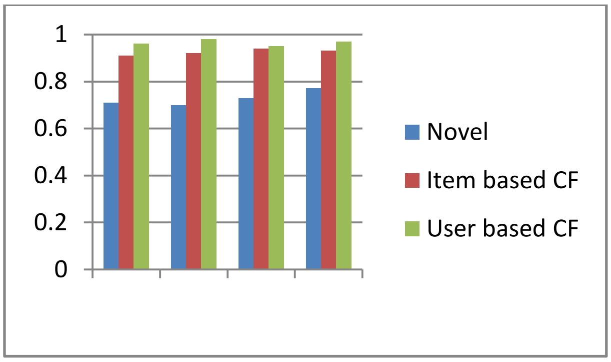


Fig 4: RMSE (MovieLens-1M dataset)

6. CONCLUSION AND FUTURE WORK

In this work, a novel model has been proposed which systematically unite contextual information with the social popularity base model to improve the quality of recommendations. Novel model divides the original rating matrix based on different contexts using random decision trees algorithm. The created sub matrix holds ratings with similar contexts thus enforcing higher impact on each other. Matrix factorization is used to the sub-matrix to predict the missing ratings. In order to efficiently include social popularity information, novel model establishes an additional social regularization term to suppose a user's preference for an item by learning his/her friends' tastes. Experiments carried on two real datasets show that novel model clearly outperforms the state-of-the-art context-aware and social recommendation models. Moreover, even if in some cases where social popularity information is not available, novel model still outperforms other context-aware approaches by efficiently forming and integrates different contextual information's.

In the future work, novel model will be applied to some real world application scenarios. Different way to incorporated social popularity information will be evaluated in future.

7. REFERENCES

[1] N. N. Liu, L. He, and M. Zhao. Social temporal collaborative ranking for context aware movie recommendation. ACM Transactions on Intelligent Systems and Technology, 4(1), 2013.

[2] Xin Liu and Karl Aberer, SoCo: A Social Network Aided Context-Aware Recommender System. In Proceedings of the International World Wide Web Conference Committee (IW3C2), 2013, Rio de Janeiro, Brazil. ACM 978-1-4503-2035-1/13/05.

[3] E. Zhong, W. Fan, and Q. Yang. Contextual collaborative filtering via hierarchical matrix factorization. In Proceedings of the SIAM International Conference on Data Mining, 2012.

[4] A. Akther, K. Alam, H.-N. Kim and A. El Saddik. Social network and user context assisted personalization for recommender systems. In Innovations in Information Technology (IIT), 2012 International Conference on, 2012.

[5] B. Xu, J. Bu, C. Chen, and D. Cai. An exploration of improving collaborative recommender systems via user-item subgroups. In Proceedings of the 21st international conference on World Wide Web, 2012.

[6] X. Yang, H. Steck, and Y. Liu. Circle-based recommendation in online social networks. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, 2012.

[7] M. Jiang, P. Cui, R. Liu, Q. Yang, F. Wang, W. Zhu, and S. Yang. Social contextual recommendation. In Proceedings of the 21st ACM international conference on Information and knowledge management, 2012.

[8] G. Gonzalez, J. L. de la Rosa, M. Montaner, and S. Delfin. Embedding emotional context in recommender systems. In Data Engineering Workshop, 2007 IEEE 23rd International Conference on, pages 845–852. IEEE, 2007

[9] Adomavicius, G., Sankaranarayanan, R., Sen, S. & Tuzhilin, A. 2005. Incorporating Contextual Information in Recommender Systems Using a Multidimensional Approach, ACM Transactions on Information Systems , 23(1), 103-145

[10] M. Buchin, S. Dodge, and B. Speckmann. Context-aware similarity of trajectories. In Geographic Information Science, volume 7478 of Lecture Notes in Computer Science, pages 43–56. Springer Berlin Heidelberg, 2012.

[11] E. Zhong, W. Fan, J. Wang, L. Xiao, and Y. Li. Comsoc: adaptive transfer of user behaviors over composite social network. In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, 2012.

[12] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King. Recommender system with social regularization. In Proceedings of the 4th ACM International Conference on Web Search and Data Mining (WSDM) , pages 287–296, 2011

[13] R. Gemulla, E. Nijkamp, P. J. Haas, and Y. Sismanis. Large-scale matrix factorization with distributed stochastic gradient descent. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, 2011.

[14] L. Baltrunas, B. Ludwig, and F. Ricci. Matrix factorization techniques for context aware recommendation. In Proceedings of the fifth ACM conference on Recommender systems, 2011.

[15] S. Rendle, Z. Gantner, C. Freudenthaler, and L. Schmidt-Thieme. Fast context-aware recommendations with factorization machines. In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, 2011.

[16] F. Ricci, L. Rokach, B. Shapira, and P. Kantor. Recommender Systems Handbook. Springer, 2011.

[17] K. Zhou, S.-H. Yang and H. Zha. Functional matrix factorizations for cold-start recommendation. In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, 2011.

[18] G. Adomavicius and A. Tuzhilin. Context-aware recommender systems. In Recommender Systems Handbook, pages 217–253. 2011.

[19] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King. Recommender systems with social regularization. In Proceedings of the fourth ACM international conference on Web search and data mining, 2011.

[20] A. Karatzoglou, X. Amatriain, L. Baltrunas, and N. Oliver. Multiverse recommendation: n-dimensional tensor factorization for context-aware collaborative filtering. In Proceedings of the fourth ACM conference on Recommender systems, 2010.

[21] N. N. Liu, B. Cao, M. Zhao, and Q. Yang. Adapting neighborhood and matrix factorization models for context aware recommendation. In Proceedings of the Workshop on Context-Aware Movie Recommendation, 2010.

[22] F. Liu and H. J. Lee. Use of social network information to enhance collaborative filtering performance. Expert Systems with Applications, 37(7):4772 – 4778, 2010.

[23] X. Su and T. M. Khoshgoftaar. A survey of collaborative filtering techniques. Advances in Artificial Intelligence, 2009, Jan. 2009.

[24] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. Computer, 42(8):30–37, Aug. 2009.

[25] H. Ma, H. Yang, M. R. Lyu, and I. King. Sorec: social recommendation using probabilistic matrix factorization. In Proceedings of the 17th ACM conference on Information and knowledge management, 2008.