Dynamic Personalized Recommendation Algorithm on Sparse Data

B.Prasanth[1] R.Latha[2]

Final Year MCA Student, VelTech HighTech Engineering College, Chennai, India

Assistant Professor, Department of MCA, VelTech HighTech Engineering College, Chennai, India

ABSTRACT: Recommender systems suggest people items or services of their interest and proved to be an important solution to information overload problem. The big problem of collaborative filtering is its. In order to solve scalability problem, we can implement the Collaborative Filtering algorithm on the cloud computing platform. Recommendation systems are very important in the fields of E-commerce and other Web-based services. One of the main difficulties is dynamically providing high-quality recommendation on sparse data. In this paper, a novel dynamic personalized recommendation algorithm is proposed, in which information contained in both ratings and profile contents are utilized by exploring latent relations between ratings, a set of dynamic features are designed to describe user preferences in multiple phases, and finally a recommendation is made by adaptively weighting the features. Experimental results on public datasets show that the proposed algorithm has satisfying performance.

KEY WORDS: Recommender System, Collaborative Filtering, Dynamic Recommendation,   Dynamic Features, Multiple Phases of Interest.

I.INTRODUCTION

The Internet provides an unparalleled opportunity for organizations to deliver digital content to their visitors instantaneously. Content consumers usually have short attention span, while possibly a large number of content venders. Nowadays the internet has become an indispensable part of our lives, and it provides a platform for enterprises to deliver information about products and services to the customers conveniently. As the amount of this kind of information is increasing rapidly, one great challenge is ensuring that proper content can be delivered quickly to the appropriate customers. Personalized recommendation is a desirable way to improve customer satisfaction and retention.

There are mainly three approaches to recommendation engines based on different data analysis methods, i.e., rule-based, content-based and collaborative filtering. Among them, collaborative filtering (CF) requires only data about past user behavior like ratings, and its two main approaches are the neighborhood methods and latent factor models. The neighborhood methods can be user-oriented or item-oriented. They try to find like-minded users or similar items on the basis of co-ratings, and predict based on ratings of the nearest neighbors. Latent factor models try to learn latent factors from the pattern of ratings using techniques like matrix factorization and use the factors to compute the usefulness of items to users. CF has made great success and been proved to perform well in scenarios where user preferences are relatively static.In most dynamic scenarios, there are mainly two issues that prevent accurate prediction of ratings – the sparsity and the dynamic nature. Since a user could only rate a very small proportion of all items, the $U \times I$ rating matrix is quite sparse and the amount of information for estimating a candidate rating is far from enough. While latent factor models involve most ratings to capture the general taste of users, they still have difficulties in catching up with the drifting signal in dynamic recommendation because of sparsity, and it is hard to physically explain the reason of the involving. The dynamic nature decides that users' preferences may drift over time in dynamic recommendation, resulting in different taste to the items in different phases of interest, but it is not well studied in previous studies. In our experiences, the interest cycle differs from user to user, and the pattern how user preferences changes cannot be precisely described by several simple decay functions. Moreover, CF approaches usually accounted the cold-start problem which is amplified in the dynamic scenario since the rate of new users and new items would be high.

Some researchers have previously attempted to solve the above problems. Hybrid approaches which combine content based and collaborative filtering in different ways were proposed to alleviate the sparsity problem where more information were mined than just in each of them. A classified item into many categories using content information and chose recent categories to perform Item-Based Collaborative Filtering (IBCF). An introduced group similarity by clustering and used it to modify original item-item similarity matrix. Some approaches emphasize utilization of time information to deal with the dynamic nature. The proposed to model temporal dynamics to separate transient factors from lasting mones.

In this paper, we present a hybrid dynamic recommendation method. First we use more information while keeping the data consistency; we use user profiles and item contents to extend the co-rate relations between ratings through each element of users, as show in Fig.1. The ratings can reflect similar users' preferences and provide useful information for recommendation. Correspondingly, in order to enable the algorithm to maintain the changing of signals quickly and to be updated conveniently, based on time series analysis(TSA) technique a set of dynamic features are proposed, and relevant ratings in each phase of inter- est are added up by applying TSA to describe user's preferences and item's reputations. Then we proposed a personalized recommendation algorithm by adaptively weighting. The result of the proposed algorithm is effective with dynamic data and per- forms better than the previous algorithms.
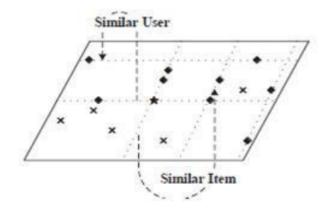

Figure1.

II.RELATED WORK

Dynamic recommendation in traditionally RMSE evaluations (even for the Netflix competition), training and testing data are randomly sampled and the train and test split is not based on time. This would produce current prediction based on future data. Even if it is guaranteed that testing instances of each user/item come later than its training instances, the aforementioned issue still exists in algorithms like IBCF and latent factor models due to the utilization of other users' future ratings. The CF approaches usually accounted the cold-start problem which is amplified in the dynamic scenario since the rate of new users and new items would be high. Some researchers have previously attempted to solve the above problems. Hybrid approaches which combine content based and collaborative filtering in different ways were proposed to alleviate the sparsely problem where more information were mined than just in each of them. et al. classified items into many categories using content information and chose recent categories to perform Item-Based Collaborative Filtering (IBCF). Kim and Li introduced group similarity by clustering and used it to modify original item-item similarity matrix

Ontology based Recommender System: In the peer to peer network (P2P network) is based on decentralized architecture has the progress of ontology based recommender system. This is basically works with dynamically changing large scale environment. In a ontology based multilayered semantic social network, is introduced. This model works on a set of users having similar interest and the correlation at different semantic levels.

Collaborative Tagging based Recommender System: In the collaborative tagging based recommender allows users particularly consumers to freely connect tags or keywords to data contents. In a generic model of collaborative tagging to recognize the dynamics behind it. The tag based system suggests the use of high quality tags, by which spam and noise can be avoided.

Dynamic Content: We consider not only the item set undergoes insertions and deletions frequently, but also the content value and then the appraisement from users are changing rapidly as well. For example, the lifetime of breaking news on the Internet is usually a couple of hours, and the value of the news (such as click through rate) is decaying temporally as people get to know it. Traditional recommender systems usually treat users' feedback static, so that feedback on the same items given at different time stamps is still comparable. This assumption doesn't hold on dynamic content. Rebuilding the model on very recent data is typically an expensive task, and tends to lose long-term interests of users. On dynamic content, recommender systems always face the cold start problem for new items.

Rule based content: Rule-based filtering creates a user-specific utility function and then applies it to the items under consideration. This approach is closely related to customization, which requires users to identify themselves, configure their individual settings, and maintain their personalized environment over time. It is easy to fail since the burden of responsibility falls on the users..

III.EXISTING METHOD

The neighborhood methods can be user-oriented or item-oriented. They try to find likeminded users or similar items on the basis of co-ratings, and predict based on ratings of the nearest neighbors. While latent factor models involve most ratings to capture the general taste of users, they still have difficulties in catching up with the drifting signal in dynamic recommendation because of sparse, and it is hard to physically explain the reason of the involving. In our experiences, the interest cycle differs from user to user, and the pattern how user preferences changes cannot be precisely described by several simple decay functions. Moreover, CF approaches usually accounted the cold-start problem which is amplified in the dynamic scenario since the rate of new users and new items would be high.

Limitations of Existing Methods:

1.   Hybrid approaches which combine content based and collaborative filtering in different ways were proposed to alleviate the sparsity problem where more information were mined than just in each of them.
2.   The principle of utilization of rating data in these algorithms some approaches emphasize utilization of time information to deal with the dynamic nature.
3.   The involved ratings can reflect similar users' preferences and provide useful information for recommendation.

IV .PROPOSED METHOD

Use only historical data but not future data for current prediction in real applications. In traditional RMSE evaluations training and testing data are randomly sampled and the train and test split is not based on time. This would produce current prediction based on future data. The data in different phases of interest have different training ratios. It is clear that the proposed algorithm is quite robust in the phases, and we found it is not true that the more recent ratings should have heavier weights across the whole time, which illustrates the advantages of the features – light computation, flexibility and high accuracy.

The Proposed Method is to m a k e use of profiles to extend the co-rating relation, and then we pro- pose a set of dynamic features to reflect user's preferences or item's reputations in different phases of interest, and after that we recommend an adaptive algorithm for dynamic personalized recommendation.

1. **Relation Mining of Rating Data**: The main complexity of capturing user's dynamic preferences is the lack of useful information, which may come from three sources - user profiles, item profiles and historical rating records during the sparsity of recommendation data. Existing algorithms mainly rely on the co-rate relation. But this will not efficient in calculation while the data is sparse as it limits the amount of data during prediction. So, to overcome this we introduce a semi co-rate relation for finding useful ratings for dynamic personalized recommendation.

2. **Dynamic Feature Extraction**: To compute better recommendation algorithm, three kinds of methods were proposed such as instance selection, time- window (usually time decay function) and ensemble learning. This technique contains a set of dynamic features to describe users' multi-phase preferences in consideration of computation, accuracy and flexibility.

3. **Adaptive Weighting Algorithm**: The parameters are quantified in the feature extraction as per the previous step, so now it's easy to organize them for accurate rating estimation by using adaptive weighting. Sizes of all the relevant subsets are also computed in MPD (Multiple Phase Division) and could reflect on data density.

4. **Root-Mean-Square Error** (RMSE), is used to evaluate the proposed recommendation algorithm. In traditional RMSE evaluation, training and testing data are randomly sampled which is not based on time. So, it would result in current prediction based on future data. Hence, Replay-match evaluation has been proposed to address this issue by Li et al whose evaluation results are more stable for dynamic recommendation.

1) To evaluate the accuracies of above mentioned dynamic recommendation algorithms as follows:
2) Sort the complete dataset in natural time order, use a certain training ratio to determine its corresponding splitting.
3) Use the previous part as the training set to adjust all pa- rameters.
4) Run algorithm on this testing set and generate estimated rating for each user-item pair.
5) Compare each estimated ratings and real ratings with in the testing set and calculate RMSE for them.
6) Use variety of ratios and cycle through the last four steps.

V.CONCLUSION AND FUTURE DIRECTIONS

In this paper, we proposed a novel dynamic personalized recommendation algorithm for sparse data, in which more rating data is utilized in one prediction by involving more neighboring ratings through each attribute in user and item profiles. A set of dynamic features are designed to describe the preference information based on TSA technique, and finally a recommendation is made by adaptively weighting the features using information in different phases of interest. The proposed algorithm is highly effective, and its computational cost is much acceptable

REFERENCES

[1] B. M. Sarwar, G. Karypis, J. A. Konstan, J. Riedl, Item-based collaborative filtering recommendation algorithms, in: WWW, 2001, pp. 285–295.
[2] P. Brusilovsky, A. Kobsa, W. Nejdl (Eds.), The Adaptive Web, Methods and Strategies of Web Personalization, Lecture Notes in Computer Science, Springer, 2007.
[3] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions, IEEE Trans. Knowl. Data Eng. 17 (6) (2005) 734–749.
[4] Y. Koren, Collaborative filtering with temporal dynamics, Communications of the ACM 53 (4) (2010) 89–97.
[5] L. Candillier, F. Meyer, M. Boull´e, Comparing state-of-the-art collaborative filtering systems, in: P. Perner (Ed.), MLDM, Vol. 4571 of Lecture Notes in Computer Science, Springer, 2007, pp. 548–562.
[6] K. Yu, A. Schwaighofer, V. Tresp, X. Xu, H. Kriegel, Probabilistic memory-based collaborative filtering, IEEE Transactions on Knowledge and Data Engineering 16 (1) (2004) 56–69.
[7] F. Fouss, A. Pirotte, J. Renders, M. Saerens, Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation, IEEE TKDE 19 (3) (2007) 355–369.
[8] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Computer 42 (8) (2009) 30–37.

[9] S. Boutemedjet, D. Ziou, Long-term relevance feedback and feature selection for adaptive content based image suggestion, Pattern Recognition 43 (12) (2010) 3925–3937.

[10] B. M. Kim, Q. Li, C. S. Park, S. G. Kim, J. Y. Kim, A new approach for combining content-based and collaborative filters, J. Intell. Inf. Syst. 27 (1) (2006) 79–91.

[11] G. Prassas, K. C. Pramataris, O. Papaemmanouil, Dynamic recommendations in internet retailing, in: ECIS, 2001.

[12] S. Rendle, C. Freudenthaler, L. Schmidt-Thieme, Factorizing personalized markov chains for next-basket recommendation, in: Proceedings of the 19th WWW, ACM, 2010, pp. 811–820.

[13] C. Xia, X. Jiang, S. Liu, Z. Luo, Z. Yu, Dynamic item-based recommendation algorithm with time decay, in: ICNC, IEEE, 2010, pp. 242–247.

[14] J. Lai, T. Huang, Y. Liaw, A fast k-means clustering algorithm using cluster center displacement, PR 42 (11) (2009) 2551–2556.

[15] A. Tsymbal, The problem of concept drift: definitions and related work, Computer Science Department, Trinity College Dublin.

[16] X. Tang, C. Yang, J. Zhou, Stock price forecasting by combining news mining and time series analysis, in: Web Intelligence, IEEE, 2009, pp. 279–282.

[17] J. Mohammed, Real-time implementation of an efficient rls algorithm based on iir filter for acoustic echo cancellation, in: IEEE/ACS ICCSA, IEEE, 2008, pp. 489–494.

[18] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, Least angle regression, The Annals of statistics 32 (2) (2004) 407–499.

[19] B. Boser, I. Guyon, V. Vapnik, A training algorithm for optimal margin classifiers, in: Proceedings of the fifth annual workshop on Computational learning theory, ACM, 1992, pp. 144–152.

[20] L. Li, W. Chu, J. Langford, X. Wang, Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms, in: Proceedings of the fourth ACM international conference on Web search and data mining, ACM, 2011, pp. 297–306.

[21] Y. Koren, Factorization meets the neighborhood: a multifaceted collaborative filtering model, in: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2008, pp. 426–434.