University College of Takestan

Volume 2,Issue 2

Available online at **http://UCTjournals.com**

UCT Journal of Research in Science ,Engineering and Technology

UCT . J. Resea. Scien. Engineer.Techno.(UJRSET)

2(2): ***52-55(2014)***

ISSN 2382-977X

Fuzzy dynamic tensor decomposition algorithm for recommender system

Mahdi Nasiri*[1], Behrouz Minaei[2], Mansour Rezghi[3]

[1.2.3.]Department of Computer, Science and Technology,Tehran, Iran

*Corresponding author's E-mail: nasiri_m@iust.ac.ir

ABSTRACT

Model base collaborative filtering has been best method in recommender system. One of the best algorithms in it is matrix and tensor decomposition which have better result for rating prediction. In this paper we propose a new tensor decomposition method based on HoSVD algorithm that use time as independent dimension. Using time in recommender systems shows sequence of user interests better. Our method utilizes rating prediction based on previous ratings. Another innovation of it is time discretion using fuzzy method. Because idea of users have low difference in near time, we fuzzify discretion of time. Results show that fuzzy discretion in deed of crisp has better results.

Original Article:

Received 21 Mar. 2014
Accepted 15 May. 2014
Published 30 Jun. 2014

Keywords:
dynamic recommender system,
collaborative filtering,
tensor decomposition,
fuzzy discretion,
sparse data

1.Introduction

One of the important problem in business intelligence is recommender items to costumers. In matrix of user-item, that show rating of users on items, we have some challenges which are interestingness problems for researchers [24]. Collaborative filtering is divided to two method –memory base and model base-. One of the important challenge of recommender systems is sparsity [7]. Most of memory base algorithms have challenge because of sparsity and usually can't solve recommender problem. Versus memory base algorithms, model base algorithms present solving methods. Among them, the algorithms which do sparsity challenge better, have better results.Sparsity of data is missing value and we must via recommender algorithms detect these missing values [9].In preprocessing of data, some algorithms transform missing values to values like 0, mean of all ratings, sum of mean of all ratings, mean of users ratings, mean of items ratings and …, or other values. Some algorithms have did on affective variables on recommender and have used them for change with missing value in preprocessing. When we can don't have missing value, we can use any recommender algorithm without considering missing values. In these methods, we have new challenge and it is noise. Attention of algorithms is find best formula among mixture of variables that have less error and noise. After preprocessing, we will use algorithms for correction of missing values. We use time as independent dimension and discrete it based on fuzzy logic because absolute discrete may has negative affect. For example, rating on an item in last day of month and first day of next month usually hasn't difference but if we discrete time in month durations, they are in different duration.

Matrix factorization (MF) has become very popular in recommender systems both for implicit and explicit feedback. Matrix factorization algorithms are one of the model base algorithms that have less error versus other algorithms [9-13].Factorization models is based on two method [24]. First type of these models factorize matrix to 2 matrix which are latent factors of main matrix dimensions. Then multiplying these factorization matrix will get main matrix.

Second method for using matrix decomposition use optimization method. In these methods we set decomposition matrixes contents as variables and present an optimization method which do minimum difference between main matrix R and multiplying matrix [16,20].

Rendle and etl. Present a method that get relationship among variables base time via marcov chain model [14].Another research of them that has used time has presented in [15]. They also use tensor decomposition method for tag recommender [6,12] and movie recommender [8,17].Koren has presented a method base on SVD algorithm for matrix decomposition [18]. After in [22] use time durable in optimization formula.All algorithms that use time affects, don't use time as independent dimension

[1, 19]. In some algorithm using time only for considering rating in one duration [11]. In other work, they add mean of rating in special time duration for effect on other rating which score on it. These method don't attend to previous or next rating [15, 22].

2.Proposed algorithm

Recommender system algorithms have many challenges. For suggest items to users or overhand, more of researchers focus on two dimension; user and item. They try to find items that users don't see them but may like them based on users and items profile. One of the problems that these methods don't scrutiny is time of rating on items via users. For example, for recommender films, in summer we may like cartoons because of our children and like comedy films in new-year holidays. Time duration can be important dimension for recommender.In this paper we use time as an independent dimension and solve a tensor with 3 dimensions that are user-item-time. For use time as independent dimension we must discrete time. If we discrete time discriminately, some ratings in end of a discrete ranges of time durations may be near day ratings of next ranges of time durations and has noise on data. Because of this reason we suggest fuzzify continues dimensions like time.

Main innovation of this paper is presentation of new method for recommender system based on fuzzy dimension and tensor decomposition. Our algorithm useHoSVD algorithm for decomposition. We present for the first time fuzzy dimension in decomposition. We solve some problemsin algorithm and recommender system like using fuzzy time dimension as independent dimension for recommender system are main innovation.After identifying fating membership for each membership function, we multiply rate and memberships. We are having a new tensor that have 3-dimension user-item-time and time is be fuzzifying. This transformation data time form crisp discretions to fuzzy discretions has another advantage in additional to better discretion that tensor is less sparse. This sparseness of data is behalf. We solve fuzzy tensor via optimization. We transform our problem to optimization function that must be minimum.

We have a tensor R that has three dimension user-item-time and rating of it because of fuzzification of time dimension divided in time memberships. For tensor R the HOSVD decomposition is written as below:

$$R = (U, I, T).S(2)$$

The expression at the right side is in fact the decomposition of tensor R to user, item and time variables that time dimension has been fuzzy. In this decomposition U, T and I are orthogonal matrices. Here also S is the tensor which plays the same role as the singular values. Below figure show decomposition of tensor which has 3 decomposed matrix user(U), Item(I) and time(T) from tensor.

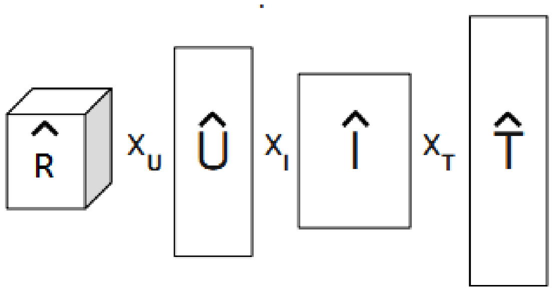

Figure 1: decomposed matrix from tensor

The question in this article is how to solve a three-dimension problem of user-item-time which time dimension be a fuzzy base discrete. In formulation of R as a tensor, we will have 3 matrix in result of tensor decomposition in which U is the user vectors matrix, I is the item vectors matrix and T is the time vectors matrix. The above formula is HOSVD algorithm which is used in its development. In fact S is a tensor which is called core. S is a tensor that have some slices. If we calculate norm of slices via any dimension of S, it's slices norm will higher the first slices and slices ranked based on their's norm. U and I and T are considered independent from oneanother. The key point here is that time is an independent fuzzy factor in this approach.

The below formula can be used for tensor decomposition R:

$$R \approx (U, I, T).S(3)$$

In which U is the user vectors matrix, I is the item vectors matrix and T is the time vectors matrix. The above formula is HOSVD algorithm which is used in its development [37, 38, 39]. In fact S is a tensor which is called core. U and I and T are considered independent from one another. The key point here is that time is an independent factor in this approach.

If q_i is equal to the row "I" of I matrix, and P_u is equal to the row "u" of the U matrix and γ_t is equal to the row "t" of the T matrix, then the recommender for "u" user in I item and "t" time by HOSVD decomposition will be as below:

$$r_{uit} = \sum_m \sum_n \sum_l (u_{um} i_{in} t_{tl}) S_{uit} (4)$$

The above formula can be reformulated and simulated as below:

$$r_{uit} = ({U^T}_u, {I^T}_i, {T^T}_t).S(5)$$

In the result of HOSVD decomposition there may be several recommender for the places we do not have any dataset of them, because we don't want the reformulated elements processed by this algorithms to be very big, same as two-dimension model. There for we must control the magnitude of decomposition matrices columns in a way that these elements do not become huge. So in tensor model, we replace the HoSVD decomposition problem with minimization problem and the result will be appropriate. The formula follows:

$$Min \sum_{(u,i,t)\in k} [(R-(U,I,T).S)+\lambda(||U_u||^2+||I_i||^2+||T_t||^2)]^2 (6)$$

In the above formula we have 2 section. The first section is square error of real rating and discriminate rating. We add second section for prevention of mutation in optimization steps. If we don't have second section we may overfitting in optimization.

We must have care that in the formula, we calculate only error of exists rating of user on item in any time. Fuzzy data has another advantage additional to previous advantage. We have less sparse in our dataset and challenge of our problem will less.

3. Result

We do this work on Movielense dataset that are rated about 7 months. We discrete this dimension in duration of month and use triangular fuzzy set for it. We are 7 months therefore 9 fuzzy membership is created. 15^{th} day of each month is center of membership and center of previous and next months are first and end of memberships. With this discretion each rating divide to two section based on rating time. For example, if we have a rating in 25'th day of 3'th month, we will have 33% of rating in 3'th membership and 66% in 4'th membership. We evaluate our method on datasets and compare fuzzy data and crisp data. We compare based on 2 methods RMSE. These formula is written as below:

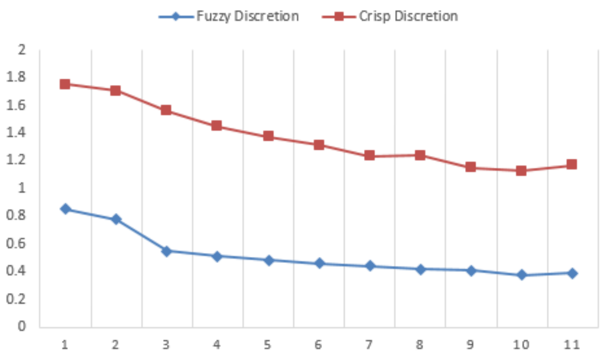

Figure 2: Comparison between fuzzy and crisp discretion of tensor decomposition

Below figure show that convergence of fuzzy discretion of time dimension has more than crisp discretion. In all number of decomposition factors, fuzzy discretion of time dimension has less RMSE than crisp discretion and more converge.Also this figure show best number of singular value which is number of hidden features. For both discretion, we have number of 10 singular value for best result. After number of 10 hidden features number of 11 and then 9 hidden features are less RMSE than others.

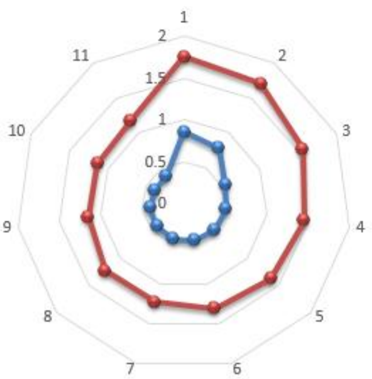

Figure 3. Compare of Crisp and Fuzzy Discretion Converge

4. Conclusion

Using time dimension in recommender system has positive effect for rating prediction. We have proof it in[3]. One problem for using time is discrete time dimension. Discretization of time if be crisp we may lose some knowledge. In this paper we present a new method that is based on tensor decomposition. In this method we use time as independent dimension. Before decompose tensor, time is discreet based on Mamdanifuzzifying. Results on datasets show that time fuzzyfing has better result in comparison with crisp discretization. Our tensor decomposition method is based on HoSVD algorithm and solve in way of optimization. In optimization we must minimize norm 2 of difference between discrimination ratings and real ratings. For future work, we can do preprocessing methods for subtraction sparsity of data. Our data if be more density, we will have better result for rating predictions. Another suggestion can be other discretization for time dimension like identifying membership of fuzzy sets.

References

[1] Chirita, P.A., Nejdl, W., zamfi, R.C., 2005. "Preventing Shilling Attacks in Online Recommender Systems", WIDM'05, pp-67-74,

[2] Esmaeili, L., Nasiri, M., Minaei-Bidgoli, B., 2011. "Analyzing Persian Social Networks: An Empirical Study", International Journal of Virtual Communities and Social Networking, pp.73-92

[3] Esmaeili, L., Nasiri, M., Minaei-Bidgoli, B., 2011. Personalizing Group Recommendation to Social Network Users, Publisher Springer-Verlag Berlin, Heidelberg, LNCS 6987, 2011, pp. 124–133.

[4] Esmaeili,L., Nasiri, M., Minaei-Bidgoli, B., 2012. "Hybrid recommender system for joining virtual communities," Journal of Applied Sciences Engineering and Technology, Vol. 5, pp. 500-509

[5] John, Z., Kush, R., Subbian, K., 2012. "Dynamic Matrix Factorization:A State Space Approachover", speech and signal processing (ICASSP)

[6] Kolda, T., Bader, B., 2009. "Tensor Decompositions and Applications". SIAM Rev. 51: 455–500

[7] Koren, Y., 2009, Collaborative Filtering with Temporal Dynamics, Proc. 15th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD 09), ACM Press, pp. 447-455

[8] Koren, Y., Bell, R., 2010. "Advanced in Collaborative filltering", Springer,USA, pp-145-186,

[9] Lathia, N., Hailes, S., Capra, L., 2009, "Temporal collaborative filtering with adaptive neighbourhoods," in Proc. ACM SIGIR Conf. Res. Dev. Inf. Retrieval, Boston, MA, Jul, pp. 796–797

[10] Minaei, B., Nasiri, M., 2014. "Convergence of Tensor Decomposition in the Recommender System based on Kullback–Leibler Divergence", Chemical and Process Engineering (CHPE)

[11] Nasiri, M., Minaei, B., Rezghi, M., 2014. "new algorithm for recommender system based on tensor decomposition", Journal of Operational Research in Its Applications, Pp:57-64

[12] Omberg, L., Golub, G., Alter, O., 2007. "A Tensor Higher-Order Singular Value Decomposition for Integrative Analysis of DNA Microarray Data From Different Studies". PNAS 104 (47): 18371–18376

[13] Osmanli. O., 2010. "A Singular Value Decomposition Approach for Recommendation Systems", PhD thesis(MSC), Middle East Technical University, pp-1-3,53-54

[14] Rendle, S., Gantner, Z., Freudenthaler, C., 2011. Schmidt-Thieme.L, "Fast Context-aware Recommendations with Factorization Machines", the 34th international ACM SIGIR conference on Research and development in information retrieval (SIGIR 2011), Beijing, China.

[15] Rezghi, M., Eld´en, L., 2011. “Diagonalization of tensors with circulantstructure”,Linear Algebra and its Applications, 422–447
[16] Salakhptdinov, R., Mnith, A., 2008. "Bayesian Probabilistic Matrix Factorization Using Marcov Chain Monte Carlo", 25th International Conference on Machine Learning, pp-880-887
[17] Salakhptdinov, R., Mnith, A., 2008. Probabilistic Matrix Factorization, Advances in Neural Information Processing Systems 20
[18] Shan, H., Banerjee, A., 2010. Generalized Probabilistic Matrix Factorization for Collaborative Filtering, ICDM, pp-1025-1030
[19] Sharifi, Z., Rezghi, M., Nasiri, M., 2013. “New algorithm for recommender systems based on singular value decomposition method” Computer and Knowledge Engineering (ICCKE), IEEE, Pp: 86-91
[20] Spiegel, S., 2010. A Hybrid Approach to Recommender Systems based on Matrix Factorization. Diploma Thesis. Berlin
[21] Takacs, G., Pilaszy, I., Nemeth, B., 2008. "Investigation of Various Matrix Factorization Method for Large Recommender Systems", 2nd Netflix-KDD Workshop,pp-553-562
[22] Takacs, G., Pilaszy, I., Nemeth, B., Tikk, D., 2008."Matrix Factorization and Neighbor Based Algorithms the Netflix Prize Problem", ACM Conference, pp-267-274
[23] Vozalis, M., Markos, A., Margaritis, K., 2010. “Collaborative Filtering through SVD-Based and Hierarchical NonLinear PCA” ICANN, VOL.6352,PP-395-400
[24] Wild, S., 2002 “Seeding non-negative matrix factorization with the spherical k-means clustering” Mas er’sesis, University