An Item Based Collaborative Filtering Using BP Neural Networks Prediction

SongJie GONG
Zhejiang Business Technology Institute,
Ningbo 315012, P. R. China
e-mail: gongsongjiezjbti@163.com

HongWu YE
Zhejiang Textile & Fashion College,
Ningbo 315211, P. R. China
e-mail: yehongwuzjbti@163.com

***Abstract*—Recommendation systems can help people to find interesting things and they are widely used in our life with the development of the Internet. Collaborative filtering technique has been proved to be one of the most successful techniques in recommendation systems in recent years. Poor quality is one major challenge in collaborative filtering recommender systems. Sparsity of source data set is the major reason causing the poor quality. Aiming at the problem of data sparsity for collaborative filtering, a new personalized recommendation approach based on BP neural networks and item based collaborative filtering is presented. This method uses the BP neural networks to fill the vacant ratings where necessary and uses item based collaborative filtering to form nearest neighborhood, and then generates recommendations. The experiment results argue that the algorithm efficiently improves sparsity of rating data, and promises to make recommendations more accurately than conventional collaborative filtering.**

Keywords-recommender system; item based collaborative filtering; BP neural networks; sparsity

I. INTRODUCTION

While the rapid growth and wide application of the Internet and information technology has provided an unprecedented abundance of information resources, it has also led to the problem of information overload. However, our experiences and knowledge often do not enough to process the vast amount of usable information [1, 2]. Thus, methods to help find resources of interest have attracted much attention from both researchers and vendors. To deal with the problem, the personalized recommendation systems play a more important role and collaborative filtering (CF) has proved to be one of the most effective for its simplicity in both theory and implementation [3, 4].

Collaborative filtering has been successfully used in various applications. The famous electronic commerce website Amazon and CD-Now have employed CF technique to recommend products to customers and it has improved quality and efficiency of their services. The CF assumes that a good way to find a certain user's interesting content is to find other people who have similar interests with him [5, 6]. CF methods operate upon user ratings on observed items making predictions concerning users' interest on unobserved items. The sparsity of ratings problem is particularly important in domains with large or continuously updated list of items as well as a large number of users. The sparsity problem may occur when either none or few ratings are available for the target user, or for the target item that prediction refers to, or for the entire database in average. Different treatments are required and different prediction techniques must be employed depending on the sparsity conditions, making the selection of an appropriate approach a cumbersome task. Current CF approaches are limited in the sense that they address specific aspects of the above problem.

Aiming at the problem of data sparsity for collaborative filtering, in this paper, a new personalized recommendation approach based on BP neural networks and item based collaborative filtering is presented. This method uses the BP neural networks to fill the vacant ratings where necessary and uses item based collaborative filtering to form nearest neighborhood, and then generates recommendations. The experiment results argue that the algorithm efficiently improves sparsity of rating data, and promises to make recommendations more accurately than conventional collaborative filtering.

II. ITEM BASED COLLABORATIVE FILTERING USING BP NEURAL NETWORKS

A. BP neural networks

BP neural network consists of an input layer, one or more hidden layer and an output layer. It has been proved from theory that constructing a 3-layer neural network is enough for approximating any nonlinear function. Thus, a typical BP neural network consists of three layers [7, 8]. The BP network is based on the supervised procedure and the network constructs a model based on examples of data with known output. Given a training set $\{ (x (t) , y (t)) \mid x (t) \in Rm , y (t) \in Rn , t = 1 ,2 , \cdots k\}$, BP can implement high nonlinear mapping from input to output. There exists a mapping F: Rm $\rightarrow$Rn such that F(x (t)) = y (t) .The BP algorithm is carried out as follows:

$$h_j(t) = f(\sum_{i=1}^{m} w_{ji} x_i(t) + a_j),\ j = 1, 2, \ldots, p$$

$$y_l^*(t) = f(\sum_{j=1}^{p} v_{lj} h_j(t) + b_l), l = 1, 2, \ldots, n$$

where f() is activation function which selected sigmoid function $f (t) = (1 + e^{-t})^{-1}$, j and l are the number of neurons

978-0-7695-3618-7/09 $25.00 © 2009 IEEE
DOI 10.1109/IIS.2009.69

of hidden layer and output layer respectively , x(t) = (x1 (t) , x2 (t) , ···, xm (t)) is input vector , hj(t) is the output of the jth neuron of hidden layer , y^*_l(t) is the lth output of the network , wji and vlj are weight s between the input/ hidden layers and hidden/ output layers , respectively , aj and bl are the bias of neurons.

B. Filling vacant ratings

We use a 3-layer neural network as figure 1. For the user-item matrix, we will predict the s user as user.s to the t item as item.t.

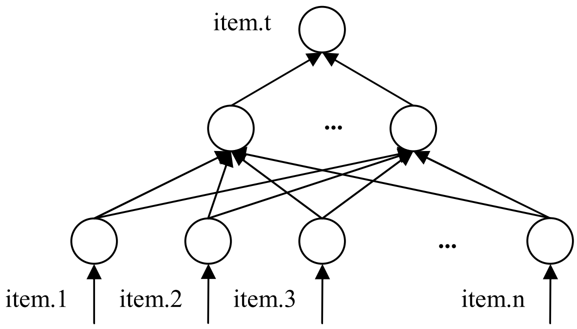

Figure 1. Structure of the BP neural network used to fill the vacant ratings

Let yl (t) be expected output of neural network. There is an error between actual output and expected output, this error, named mean square error (MSE), can be expressed by the function:

$$E = \frac{1}{2}\sum_{t=1}^{k}\sum_{l=1}^{n}(y_l(t) - y_l^*(t))^2$$

BP algorithm is a gradient descent algorithm, in which the network weights are moved along the negative of the gradient of the MSE function. Input vectors and the corresponding target vectors are used to train the network repeatedly until the error reaches the satisfaction. The trained BP neural network have the ability to generalize, that is, once trained, the system is able to process previously unseen data sample and to yield a probable response.

C. Measuring the item rating similarity

There are several similarity algorithms that have been used[5,6,9]: Pearson correlation, cosine vector similarity, adjusted cosine vector similarity, mean-squared difference and Spearman correlation.

Pearson's correlation, as following formula, measures the linear correlation between two vectors of ratings as the target item t and the remaining item r.

$$sim(t,r) = \frac{\sum_{i=1}^{m}(R_{it} - A_t)(R_{ir} - A_r)}{\sqrt{\sum_{i=1}^{m}(R_{it} - A_t)^2\sum_{i=1}^{m}(R_{ir} - A_r)^2}}$$

Where R_{it} is the rating of the target item t by user i, R_{ir} is the rating of the remaining item r by user i, A_t is the average rating of the target item t for all the co-rated users, A_r is the average rating of the remaining item r for all the co-rated users, and m is the number of all rating users to the item t and item r.

The cosine measure, as following formula, looks at the angle between two vectors of ratings as the target item t and the remaining item r.

$$sim(t,r) = \frac{\sum_{i=1}^{m} R_{it}R_{ir}}{\sqrt{\sum_{i=1}^{m} R_{it}^2\sum_{i=1}^{m} R_{ir}^2}}$$

Where R_{it} is the rating of the target item t by user i, R_{ir} is the rating of the remaining item r by user i, and m is the number of all rating users to the item t and item r.

The adjusted cosine, as following formula, is used for similarity among items where the difference in each user's use of the rating scale is taken into account.

$$sim(t,r) = \frac{\sum_{i=1}^{m}(R_{it} - A_i)(R_{ir} - A_i)}{\sqrt{\sum_{i=1}^{m}(R_{it} - A_i)^2\sum_{i=1}^{m}(R_{ir} - A_i)^2}}$$

Where R_{it} is the rating of the target item t by user i, R_{ir} is the rating of the remaining item r by user i, A_i is the average rating of user i for all the co-rated items, and m is the number of all rating users to the item t and item r.

D. Producing recommendations

Since we have got the membership of item, we can calculate the weighted average of neighbors' ratings, weighted by their similarity to the target item.

The rating of the target user u to the target item t is as following:

$$P_{ut} = \frac{\sum_{i=1}^{c} R_{ui} \times sim(t,i)}{\sum_{i=1}^{c} sim(t,i)}$$

Where Rui is the rating of the target user u to the neighbour item i, sim(t, i) is the similarity of the target item t and the neighbour item i, and c is the number of the neighbours.

III. DATASET AND MEASUREMENT

A. Data set

We use MovieLens collaborative filtering data set to evaluate the performance of proposed algorithm. MovieLens data sets were collected by the GroupLens Research Project at the University of Minnesota and MovieLens is a web-based research recommender system that debuted in Fall 1997. Each week hundreds of users visit MovieLens to rate and receive recommendations for movies [9,10]. The site now has over 45000 users who have expressed opinions on 6600 different movies. The ratings are on a numeric five-point scale with 1 and 2 representing negative ratings, 4 and 5 representing positive ratings, and 3 indicating ambivalence.

B. Performance measurement

Several metrics have been proposed for assessing the accuracy of collaborative filtering methods. They are divided into two main categories: statistical accuracy metrics and decision-support accuracy metrics. In this paper, we use the statistical accuracy metrics [11,12].

Statistical accuracy metrics evaluate the accuracy of a prediction algorithm by comparing the numerical deviation of the predicted ratings from the respective actual user ratings. Some of them frequently used are mean absolute error (MAE), root mean squared error (RMSE) and correlation between ratings and predictions. All of the above metrics were computed on result data and generally provided the same conclusions. As statistical accuracy measure, mean absolute error (MAE) is employed.

Formally, if n is the number of actual ratings in an item set, then MAE is defined as the average absolute difference between the n pairs. Assume that p1, p2, p3, ..., pn is the prediction of users' ratings, and the corresponding real ratings data set of users is q1, q2, q3, ..., qn. See the MAE definition as following:

$$MAE = \frac{\sum_{i=1}^{n} | p_i - q_i |}{n}$$

The lower the MAE, the more accurate the predictions would be, allowing for better recommendations to be formulated. MAE has been computed for different prediction algorithms and for different levels of sparsity.

C. Comparing with the traditional collaborative filtering

We compare the proposed method with the traditional collaborative filtering. The performance of our proposed approach using BP neural networks is better than the traditional collaborative filtering in terms of the MAE measure.

IV. CONCLUSIONS

Recommendation systems can help people to find interesting things and they are widely used in our life with the development of the Internet. Collaborative filtering technique has been proved to be one of the most successful techniques in recommendation systems in recent years. Poor quality is one major challenge in collaborative filtering recommender systems. Sparsity of source data set is the major reason causing the poor quality. Aiming at the problem of data sparsity for collaborative filtering, in this paper, a new personalized recommendation approach based on BP neural networks and item based collaborative filtering is presented. This method uses the BP neural networks to fill the vacant ratings where necessary and uses item based collaborative filtering to form nearest neighborhood, and then generates recommendations. The experiment results argue that the algorithm efficiently improves sparsity of rating data, and promises to make recommendations more accurately than conventional collaborative filtering.

REFERENCES

[1] Kwok-Wai Cheung, LilyF. Tian, Learning User Similarity and Rating Style for Collaborative Recommendation, Information Retrieval, 2004, 7, 395–410,

[2] Tae Hyup Roh, Kyong Joo Oh, Ingoo Han, The collaborative filtering recommendation based on SOM cluster-indexing CBR, Expert System with Application 25(2003) 413-423.

[3] Manos Papagelis, Dimitris Plexousakis, Qualitative analysis of user-based and item-based prediction algorithms for recommendation agents, Engineering Application of Artificial Intelligence 18 (2005) 781-789.

[4] Yu Li, Liu Lu, Li Xuefeng, A hybrid collaborative filtering method for multiple-interests and multiple-content recommendation in E-Commerce, Expert Systems with Applications 28 (2005) 67–77.

[5] Shih, Y.-Y., & Liu, D.-R., Product recommendation approaches: Collaborative filtering via customer lifetime value and customer demands, Expert Systems with Applications (2007),

[6] Bo Xie, Peng Han, Fan Yang, Rui-Min Shen, Hua-Jun Zeng, Zheng Chen, DCFLA: A distributed collaborative-filtering neighbor-locating algorithm, Information Sciences 177 (2007) 1349–1363.

[7] Zhang Feng, Chang Huiyou,Employing BP Neural Networks to Alleviate the Sparsity Issue in Collaborative Filtering Recommendation Algorithms, Journal of Computer Research and Development, 43 (4) : 667～672 , 2006.

[8] HE Fangguo, Qi Huan, Back propagation neural network based on modified genetic algorithm and its application, Journal of huazhong normal university (Nat. Sci.), Vol. 41 No. 1 Mar. 2007:51-54.

[9] Sarwar B, Karypis G, Konstan J, Riedl J. Item-Based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International World Wide Web Conference. 2001. 285-295.

[10] Huang qin-hua, Ouyang wei-min, Fuzzy collaborative filtering with multiple agents, Journal of Shanghai University (English Edition), 2007,11(3):290-295.

[11] Gao Fengrong, Xing Chunxiao, Du Xiaoyong, Wang Shan, Personalized Service System Based on Hybrid Filtering for Digital Library, Tsinghua Science and Technology, Volume 12, Number 1, February 2007,1-8.

[12] M.G. Vozalis, K.G. Margaritis, Using SVD and demographic data for the enhancement of generalized Collaborative Filtering, Information Sciences 177 (2007) 3017–3037.