An Effective Recommendation Algorithm for Clustering-Based Recommender Systems

Taek-Hun Kim and Sung-Bong Yang

Dept. of Computer Science, Yonsei University,
134 Shinchon-dong, Seadaemun-gu, Seoul, 120-749, Korea
{kimthun, yang}@cs.yonsei.ac.kr

Abstract. In this paper we present an effective recommendation algorithm using a refined neighbor selection and attributes information on the goods. The proposed algorithm exploits the transitivity of similarities using a graph approach. The algorithm also utilizes the attributes of the items. The experiment results show that the recommendation system with the proposed algorithm outperforms other systems and it can also overcome the very large-scale dataset problem without deteriorating prediction quality.

1 Introduction

A recommender system using collaborative filtering which we call it CF, calculates the similarity between the test customer and each of other customers who have rated the items that are already rated by the test customer. Since CF is based on the ratings of the neighbors who have similar preferences, it is very important to select the neighbors properly to improve prediction quality.

With millions of customers and items, a recommender system running on an existing algorithm will suffer seriously the scalability problem. Therefore, there are demands for a new approach that can quickly produce high quality predictions and can resolve the very large-scale problem. Clustering techniques often lead to worse prediction accuracy than other methods. Once clustering is done, however, performance can be quite good, since the size of a cluster that must be analyzed is much smaller. Therefore, the clustering-based method can solve the very large-scale problem in recommender systems[2][3][4].

2 The Clustering-Based CF

In CF, $p_{a,i}$ is used to predict the preference of a customer and computed with Equation (1) in [4]. In the Equation $w_{a,k}$ is the Pearson correlation coefficient which can be computed with Equation (2) in [4].

The k-means clustering method creates k clusters each of which consists of the customers who have similar preferences among themselves. In this method we first select k customers arbitrarily as the initial center points of the k clusters,

S. Zhang and R. Jarvis (Eds.): AI 2005, LNAI 3809, pp. 1150–1153, 2005.

respectively. Then each customer is assigned to a cluster in such a way that the distance between the customer and the center of a cluster is maximized. The Pearson correlation coefficient can be used as the distance.

If the clustering process is terminated, we choose the cluster with the highest Pearson correlation coefficient from its center to the test customer. Finally, prediction for the test customer is calculated with all the customers in the chosen cluster. The clustering-based neighbor selection method can give a recommendation quickly to the customer when the dataset is quite large, because it selects customers only from the best cluster only as the neighbors[3].

3 The Proposed Recommendation Algorithm

We propose an effective recommendation algorithm for clustering-based recommender systems. It uses a refined neighbor selection algorithm(RNSA) that considers both high and low similarities with respect to the test customer and exploits the transitivity of similarity using a graph approach. The proposed algorithm also utilizes the attributes of the items in the process of prediction for high prediction quality.

We regard a portion of the input dataset a complete undirected graph in which a vertex represents each customer and a weighted edge corresponds to the similarity between two endpoints (customers) of the edge. RNSA creates k clusters from the input dataset with the k-means clustering method. Then it finds the best cluster $\boldsymbol{C}$ with respect to the test customer $\boldsymbol{t}$ among the k clusters. RNSA adds the test customer $\boldsymbol{t}$ into the best cluster $\boldsymbol{C}$ and regard it as $\boldsymbol{v}$. RNSA then searches the unmarked vertices adjacent to $\boldsymbol{v}$ who have the similarities either larger than δ_H or smaller than δ_L, where δ_H and δ_L are some threshold values for the Pearson correlation coefficients. Note that as the threshold values changes, so does the size of the neighbors. The search is performed in a breadth-first manner. That is, we search the adjacent vertices of $\boldsymbol{v}$ according to δ_H and δ_L to find the neighbors of $\boldsymbol{t}$, and then search the adjacent vertices of each neighbor of $\boldsymbol{v}$ in turn. The search stops when we have enough neighbors for prediction. The following describes RNSA in detail. When the algorithm is terminated, the test customer $\boldsymbol{t}$ is removed from the set, Neighbors and the set is returned as output.

The Refined Neighbor Selection Algorithm
Input: the test customer $\boldsymbol{t}$, the input dataset $\boldsymbol{S}$
Output: ***Neighbors***

1. Create k clusters from $\boldsymbol{S}$ with the k-means clustering method;
2. Find the best cluster $\boldsymbol{C}$ for the test customer $\boldsymbol{t}$;
3. Add the test customer $\boldsymbol{t}$ into the best cluster $\boldsymbol{C}$ and regard it as $\boldsymbol{v}$;
4. Add $\boldsymbol{v}$ to ***Neighbors***;
5. If we have enough neighbors then return Neighbors. Otherwise, traverse $\boldsymbol{C}$ from $\boldsymbol{v}$ in a breadth-first manner when visiting vertices (customers). The sim-

ilarity of the customer is checked to see if it is either higher than δ_H or lower than δ_L. If so, let $\boldsymbol{v}$ = the customer. Go to Step 4;

Note that in Step 5 we terminate the algorithm if the number of levels (depths) we search from the test customer added in Step 3 in a breadth-first manner is greater than a fixed value. This value can be determined through various experiments. The test customer $\boldsymbol{t}$ is removed from the Neighbors before returning the set.

For using the attributes in prediction we use Equation (1) proposed by us in [4] as a new prediction formula in order to predict customer's preferences more accurately. In this equation, $A(\overline{r_{a,i}})$ and $A(\overline{r_{k,i}})$ are the averages of customer a and k's attribute values, respectively.

$$P_{a,i} = A(\overline{r_{a,i}}) + \frac{\sum_{k}\{w_{a,k} \times (r_{k,i} - A(\overline{r_{k,i}}))\}}{\sum_{k} \mid w_{a,k} \mid} \quad . \tag{1}$$

4 Experimental Results

In order to evaluate the prediction accuracy of the proposed recommendation system, we used the *MovieLens* dataset of the *GroupLens Research Center*[5]. In the *MovieLens* dataset, one of the valuable attributes of an item is the genre of the movies. And we used the mean absolute error(MAE) as the evaluation metrics. MAE is one of the statistical prediction accuracy metrics for evaluating recommender systems[1][2][4].

For the experiment, we have chosen randomly 10% out of all customers in the dataset as the test customers. For each test customer, we have chosen randomly ten movies that are actually rated by the test customer as the test movies. The final experimental results are averaged over the results of ten different test sets for a statistical significance.

We have implemented four recommendation systems for the experiments. The first one is the recommendation system only with the clustering-based CF, called R_{kcf}. The second one is R_{kcf} with the attribute information utilized, called R_{attr}. The third one is R_{kcf} with RNSA, called R_{rnsa}. And the last one is R_{kcf} with both RNSA and the attribute information, called $R_{proposed}$, which is the proposed recommendation algorithm.

The experimental results are given in Table 1. We have tested various number of clusters and have provided the results for typical numbers in multiples of 10. We have determined the threshold values which gave us the smallest MAEs through various experiments. After we have tested extensively, we have obtained that a search depth d is 2 for all the cases. It appears that having many neighbors does not necessarily contribute toward better prediction.

The results show that $R_{proposed}$ outperforms other systems for all the cases. We also found that both R_{rnsa} and R_{attr} are better than R_{kcf} and R_{rnsa} is better

Table 1. The experimental results

k	R_{kcf}	R_{attr}	R_{rnsa}	$R_{proposed}$
2	0.750977	0.700116	0.739342	0.668406
10	0.788403	0.751374	0.739447	0.668461
20	0.814820	0.793039	0.739449	0.668499
30	0.832379	0.811756	0.739440	0.668418
40	0.835119	0.822225	0.739474	0.668445
50	0.841960	0.827376	0.739488	0.668512

than R_{attr}. These fact means that the clustering-based recommender system using both refined neighbor selection and attributes can solve the very large-scale problem without deteriorating prediction quality.

5 Conclusions

In this paper we have proposed an effective recommendation algorithm that finds valuable neighbors using a graph approach along with clustering and exploiting the attribute information of the items. The experimental results show the recommendation system with the proposed algorithm outperforms other systems for all the cases. Therefore, the recommendation system with the proposed system can be a choice to resolve the very large-scale problem while it gives high prediction quality.

Acknowledgements

We thank *the GroupLens Research Center* for permitting us to use the MovieLens dataset. This work was supported by the Brain Korea 21 Project in 2005.

References

1. Jonathan L. Herlocker, Joseph A. Konstan, Al Borchers, and John Riedl: An Algorithmic Framework for Performing Collaborative Filtering. Proceedings of the 22nd International ACM SIGIR Conference on Research and Development in Information Retrieval. (1999)
2. John S. Breese, David Heckerman, and Carl Kadie: Empirical Analysis of Predictive Algorithms for Collaborative Filtering. Proceedings of the Conference on Uncertainty in Artificial Intelligence. (1998)
3. Badrul M. Sarwar, George Karypis, Joseph A. Konstan, John T. Riedle: Recommender Systems for Large-Scale E-Commerce: Scalable Neighborhood Formation Using Clustering. Proceedings of the Fifth International Conference on Computer and Information Technology. (2002)
4. Taek-Hun Kim, Sung-Bong Yang: Using Attributes to Improve Prediction Quality in Collaborative Filtering. Lecture Notes in Computer Science, Vol. 3182. (2004)
5. MovieLens dataset, GroupLens Research Center, url: http://www.grouplens.org/.