Appl. Math. Inf. Sci. 6 No. 1S pp. 135S-141S (2012)

Applied Mathematics & Information Sciences
An International Journal

@ 2012 NSP
Natural Sciences Publishing Cor.

Research on the Recommendation System in a Pure P2P Environment

Ying Gao[1] and Jiang Zhan[2]

[1]*School of Information ,Capital University of Economics and Business, Bei jing 100070,China*
Email: 1gaoying517@cueb.edu.cn
[2]*School of Information,Renmin University of China,Beijing 100872,China*
Emai: zhanjiang@ruc.edu.cn

Received Dec. 22, 2010; Revised Feb. 21, 2011; Accepted Apr. 23, 2011

Abstract: At present, most peer-to-peer file sharing systems provide file-mark-based searching function. In the paper, we offer a file recommendation system based on pure peer-to-peer system, use the TOPK method to find similar interest peers in a way to realize the interest-based self-adaptive adjustment of network topology; and exploits the recommendation information between users with similar interest to achieve the high-efficiency searching of files in the network.

Keywords: P2P, Interest Community, Recommendation System

1 Introduction

Currently, file share has become a most widely employed application of P2P technology. In a manner to facilitate users to correctly search relevant files in the network, elevate the P2P file sharing system's use efficiency to the utmost, a deluge of work has been devoted to ameliorating researches on the search performance of P2P in recent years. These researches were implemented mainly based on file key words, aimed at raising the searching effects of text files. However, people prefer to utilize P2P to share music, movies and other multimedia files [1][2]. Presently, no sound technologies are available to extract the features of non-text files, thus causing findings of key word-based researches unable to play a role in P2P file sharing systems. In this paper, we consider to merit the references of recommendation system theories, capitalize on the collaborative filtering method to recommend files to users in order to enhance the file searching efficiency in P2P file sharing systems [3][4].

This paper studies how to recommend files according to user's feedbacks in a pure peer-to-peer environment. Assuming every peer in the network maintains the assessment information of some shared files, we use Pearson Correlation Coefficient to calculate the interest similarities between peers. In a pure peer-to-peer network (like Gnutella), neighbor relationship is established randomly when a peer joins the network, for every pair of neighbors with similar interests, they may not necessarily be neighbors physically. This way, peers can not necessarily get a very good recommendation result if using the recommendation information from a neighboring peer in the network. We propose a self-adaptive adjustment method of network topology in accordance with the interests between peers, and let the peers with similar interests approximate each other as near as possible through the mutual exchange of view list between neighbors. As the recommendation goes on and the user evaluation data accumulates, the recommendation quality will be gradually upgraded. Furthermore, such work is self-adaptive; therefore the maintenance cost is quite small. Experimental results demonstrate our method can reach very sound recommendation effect, tantamount to a central server in the network, when the network access peer is only around 10%.

2 Relevant Work Comparison

Searching in the Gnutella network is a low-efficiency broadcasting-type one. To raise the searching efficiency, researchers have advanced many solutions, but mainly based on key words matching in files, without good effects for movie, music and other multimedia files. In this regard, collaborative filtering [5] is a powerful technology to handle the problem; it can utilize other users' assessments to recommend probably user-interested files.

Conventional recommendation systems mostly apply C/S structure, where a central server supplies services to many clients. The recommendation systems assemble every user's feedback, then implements local learning. Such recommendation systems perform well in quality due to the comprehensive collection of user feedbacks. Nevertheless, such recommendation systems must depend on a central service to realize efficient information gathering and management, which is impossible to materialize for pure P2P networks, such as Gnutella, famous for the no-central-server feature [6]. [7] Proposed a superpeer-based TV program recommendation system, where every superpeer has to add peers with the same interests as friends; however, the problem is every peer has to store a deluge of friends. PipeCF [8] is a DHT-based recommendation system, which map every peer's recommendation to some peer with item and scores. The basic assumption in PipeCF composes: for the harmony of two users' interests, they have to give the same marks to at least one product. First, this assumption is open to question and discussion. Secondly, in PipeCF, the expandability is poor for the complete mark data of users have to be saved in every product she/he marked, thus causing data redundancy. In the light of the situation where only few peers offer services, most peers just use services in the network, when a peer in evaluates another peer's service quality in the Gnutella, it sends an inquiry to an automatically created neighboring peer in the Gnutella, and applies the votes of neighboring peers to determine the service quality.

This paper puts forward an interest community-based recommendation system in P2P networks. In the system, peers dynamically adjust their own neighboring peers in the network according to their interests. When they need files, they only need to send an inquiry to neighboring community peers to prevent the overflow of messages, and to lift the recommendation quality. As the recommendation proceeds and the frequent data of users rises, the peer neighbors can be dynamically aligned with the characteristic of self-adaptation.

3 Collaborative Filtering in a Pure P2P Environment

Pure P2P networks (such as Gnutella) embody the following features: no central server existed, the equal position of every peer, ability to dynamic joining of exiting from the network, data put in peers up to users, random network topological structures .

The collaborative filtering recommendation creates recommendation lists for target users based on the users' opinions; it is based on such a hypothesis [5]: If users give similar points to some projects, they must give similar marks to other projects. The collaborative filtering recommendation system exploits statistical techniques to search a certain number of the nearest neighbors for target users, and then predict the target users' assessment on projects in accordance with the nearest neighbors' assessments of the projects, thus generating a corresponding recommendation list. To find the nearest neighbors of target users, it is a must to measure the similarity between users, and then choose a given quantity of users with the highest similarity as the nearest neighbors of target users. Whether the search of the nearest neighbors of target users directly relates to the recommendation quality of the entire recommendation system is vital to the success of the whole collaborative filtering recommendation. The user scoring data can employ a m×n order matrix A(m,n), where m represents the number of users, n stands for the quantity of projects, and the element Ri(in Row i Line j), j indicates user i's assessment of project j. The user scoring data matrix is shown in Fig. 1.

The method to measure the similarity between user i and user j is, first to obtain all the projects scored by user i and user j, afterwards compute the similarity between user i and user j with variant similarity measuring methods. The similarity is written as sim (i, j).

	Item1	…	Item	…	Item
User1	R1,1	…	R1,k	…	/
…	…	…	…	…	…
User	Rj,1	…	/	…	Rj,n
…	…	…	…	…	…
Userm	/	…	Rm,k	…	Rm,n

Fig. 3.1: User rating data matrix

We use Pearson Correlation to figure out the similarity sim (i, j) between peer i and peer j, the computation equation is as below:

$$sim\ (i,\ j) = \frac{\sum_{c\in I_{ij}}\left(R_{i,c}-\overline{R_i}\right)\left(R_{j,c}-\overline{R_j}\right)}{\sqrt{\sum_{c\in I_i}\left(R_{i,c}-\overline{R_i}\right)^2}\sqrt{\sum_{c\in I_j}\left(R_{j,c}-\overline{R_j}\right)^2}} \tag{3.1}$$

where $R_{i,c}$ is peer i's score on project c, $\overline{R_i}$ and $\overline{R_j}$ separately indicate the average scores of project j by peer i and peer j; I_{ij} is the collection of jointly scored projects by peer i and peer j, and the denominator is a normalized factor.

Broadcasting-type message sending is utilized in pure P2P networks. When broadcasting information, the inquiry for recommendation messages is sent to all the neighboring peers. The scope of message sending is decided by the TTL (time-to-live) of the broadcast, and when the message is resent for a time, TTL reduces 1. When TTL reaches 0, the message-sending-process stops. When a peer finds the results, it will retrieve the results to the initial peer along the original route. To prevent the advent of loop, every inquiry message has a sole SN. If the inquiry message has previously been received, it means loop happened, so there is no need to resend it.

The Gnutella-adopted message broadcast is grounded on file name query. In the collaborative filtering system, we replaced the information query in Gnutella to the assessment list or viewlist of peers initializing inquiries to gain similar peers. Each peer maintains its own viewlist, and calculates and request the similarity based on Pearson Correlation Similarity. If the similarity exceeds a threshold value, then the solicited-peer will receive the viewlist of the current peer and trust the recommendation information from the current peer.

4 Interest Community-Based Recommendation System

Broadcasting with the Gnutella-offered neighboring peers will find few neighbors with similar interests. Thus the recommendations grounded on such neighboring peers effect not ideally. Our method is to allow peers to arrange peers with the same interests discovered in the broadcast course as their own neighbors, moreover add peers outside the broadcasting range in the friend list through exchanging peerlist of the same interests with the neighbors during the recommendation process.

4.1 The Formation of Interest Community

In the algorithm of self-adaptive topological structures, a peer arranges the combo of its neighboring peers periodically according to the accumulated viewpoint in transactions, in a way to realize the self-adaptive topology building. Meanwhile, during the entry 、exit of peers as well as the completion of every transaction, topology of corresponding peers will be rearranged.

P2P network can be revealed as a directed graph G=（P，E）,where P suggests the collection of peers, E indicates the collection of edges, （i，j）signifies the continuity between peer i and peer j; i,j∈P hints the transmitting channel of peer inquiry information.

Let Nb(i) exhibit the collection of peer i's neighboring peers. Define lmini and lmaxi as the minimum value and maximum value of the permitted quantity of connected peers, and 1≤lmini≤|Nb(i)|≤lmaxi.

Define Function Minsimi(Nb(i)) represent the several peers with the lowest similarity within the collection Nb（i）.

4.2 The Selection of Top-k Neighbors

In the formation algorithm of the interest community, we hope users to designate a similarity threshold value, when the similarity between two peers surpasses the value, we can believe the two peers are neighbors.

Nonetheless in a Gnutella network, users don’t have the global knowledge, unaware of the scoring conditions of other peers, thus difficult to designate a suitable threshold value. We streamlined the formation of interest community, so users don’t have to manually designate the threshold value, and the system will provide k neighbors with the biggest similarity for peers.

When choosing k neighbors, the simplest way is to send the conditions of k peers on the retrieval route, and gradually combine them; when reaching the top level, the top-k results will be combined.

Regarding the top-k function, we use the divide-and-conquer method. Because the search process can be indicated by a query tree, we can conduct hierarchy computation of the top-k function based on the query tree, and sequence the results and combine every peer in the network, to realize the distributed top-k query. The concrete recursive computing formula is as below:

$$\text{Top-k1}(P,Q,TTL)=\text{maxk}(\text{Local-Top-k}(P,Q),$$
$$\text{Top-k1}(P1,Q,TTL-1)\ldots,\text{Top-k1}(Pn,Q,TTL-1)) \qquad (4.2)$$

$$\text{Top-k1}(P,Q,0)=\text{Local-Top-k}(P,Q) \qquad (4.3)$$

In the formula, P1,P2…,Pn indicate the neighboring peers of Peer P, and they can also be represented as the children peers in the Query Tree (P,Q,TTL). The message is broadcast to Peers P1,P2…,Pn to implement top-k query. Apart from carrying out local top-k query, Peer P also assembles the top-k results from Children Peers P1,P2…,Pn, and creates the optimized top-k results. Leaf peer (whose TTL is 0 or without neighboring peers) only achieve local top-k query to return the top-k results to its parent peer. The entire query process goes on from bottom to the top until the root peer ultimately finds the top-k results. For instance in Fig.1, Peer j searches local information and returns the local top-k results to its parent peer. Peer b aggregates the returned top-k results by Peers d, e, f and the local top-k results, in order to produce a optimal top-k results, and continually return to Peer a. Peer a integrates the returned top-results by Peer b and Peer c as well as the local top-k results, to generate the final top-k results to give back to users.

Algorithm 1: the hierarchy top-k query algorithm based on broadcast searching, broadcastTopKQuery.

```
Input: top-k query, TTL, k indicates the number of returned results, P means the peers conducting searches.
Output: the optimal top-k results of the query tree
P.broadcastTopKQuery(query,TTL,k)
begin
localResult=localTopKQuery(query,k);
if (TTL==0) then return(localResult);
for each of my neighbors Pi do
begin
 if Pi is not visited then   peerResulti= Pi.broadcast TopK Query (query,TTL-1,k);
end
overallResult=MergeResult(localResult,peerResult1, peerResult2…,peerResultn,k);
return(overallResult);
end
```

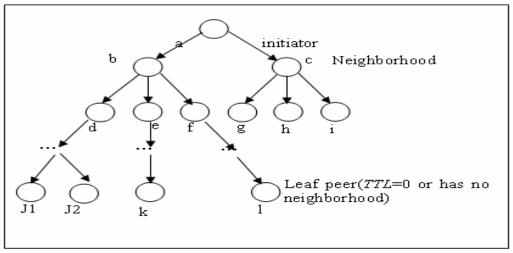


Fig. 4.1: Query Processing Tree

For example, in Figure 2 the largest similarity derives from the returned result of Peer J1; then in the second-step viewlist transmission, it first reaches Peer J1 and gets the k top-ranking viewlist, afterward combines with the viewlist of J1’s parent peer to generate a viewlist with k peers of the largest similarity. If the k viewlist’s minimum value already exceeds J2, then there is no need for J2 to transmit viewlist. If it is lower than J2, it has to reach J2 to search. The number of queries is k-(the current one which is higher than J2). According to this train of thought, it ceaselessly combines to the top, until the combination terminates.

4.3 Recommendation Algorithm based on Interest Community

The improvement of similarity computation between neighbors: we believe the more the products which have been jointly scored by users, the better the recommendations from computed similarities. Accordingly, our recommendation adopts the optimum formula:

Assuming each peer in the network has to uphold its feedback, for a given peer, we can utilize the Pearson Correlation Techniques to calculate the interest similarities between peers in the network, and apply Pearson Correlation Similarity Measuring Method to obtain the nearest neighbors of the current peer. The next step will generate corresponding recommendations. Let NBS_u stand for the collection of Peer u's nearest neighbors, then Peer u's predicted scores on Project i can be calculated through the following formula:

$$P_{u,i} = \overline{R_u} + \frac{\sum_{n \in NBS_u} sim(u,n) * \left(R_{n,j} - \overline{R_n}\right)}{\sum_{n \in NBS_u} \left(\left|sim(u,n)\right|\right)} \quad (4.3)$$

Where, $sim(u,n)$ suggests the similarity between Peer u and Peer n, $R_{n,j}$ states Peer n's score on Project j. $\overline{R_u}$ and $\overline{R_n}$ separately demonstrates the average scores of Peer u and Peer n on a project.

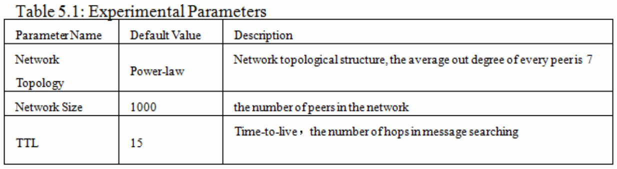
5. Experiment Result and Analysis

We verified the validity and efficiency of the interest community-based recommendation system via experiments, and implemented experimental analysis on the recommendation performance of the system. The recommendation performance is taken into account from user's perspective to reflect the system's service quality or QoS (Quality of Service); The recommendation efficiency is considered from the angle of the system to use the least resources and attain the most output.

5.1 Experimental Settings

All experiments are completed in a single PC, whose configuration is P4 1.6GHz CPU P4 1.6GHz, 1G RAM, Windows XP OS. The simulation program is written in JAVA, the network topological structure is based on Power-law generated by PLOD algorithm [11]. The study shows the network topological structure of Gnutella approximates Power-law [11], and the Internet also abides by Power-law, therefore our simulation is near to the P2P networks in actual use. And the followings are some involved parameters:

Table 5.1: Experimental Parameters



Parameter Name	Default Value	Description
Network Topology	Power-law	Network topological structure, the average out degree of every peer is 7
Network Size	1000	the number of peers in the network
TTL	15	Time-to-live，the number of hops in message searching

Note: PLOD algorithm-generated network is an undirected graph, whose average out degree of peers if 7, equivalent to have 3500 undirected sides in the undirected graph. In experiments, because we have to arrange the peer neighbors in accordance with the interests between peers, we transformed an undirected side into two directed ones. Meanwhile, in order to guarantee newly joined peers find their own community, every peer randomly selects view-lists of k peers.

5.2 Data Set

Testing data employs the data set provided by MovieLens website (http://movielens.umn.edu/). MovieLens is a Web-based research-type recommendation system, used to receive users' scores on movies and offer movie recommendation lists. At present, the website has more than 43 000 users, with 3500 movies scored by users.

In the user scoring database, we chose 6000 scores data as the experimental data set, which embodies 3000 users and 805 movies. In the data set, every user has scored at least 20 movies.

The whole experimental data set demands further division into exercise set and testing set. To that end, we introduced the variant x to represent the ratio of the exercise over the entire data set. For instance, x=0.8 indicates that 80% of the data set is exercise set, and the remaining 20% is testing set. In all experiments of the paper, x=0.8 is exploited as the experiment basis.

To measure the sparsity of the entire data set, we introduced the concept of sparsity level, defined as the percentage of unscored items by users in the user scoring data matrix. The sparsity level of the movie data set we chose is 1–6000/(145×805)=0.9486.

5.3 Measurement Scale

In a move to judge the recommendation effects of the interest community-based recommendation system, we used the Recall Ratio in information retrieval. The computation formula of Recall is listed below:

$$Re\quad call = \frac{|Ra|}{|R|} \qquad (5.1)$$

Where, $|Ra|$ is the quantity of neighbors found to meet certain threshold values using our recommendation system, $|R|$ is the number of peers satisfying conditions in the network.

Measuring scales to evaluate the recommendation quality of a recommendation system compose of two main types-statistical accuracy and decision-support accuracy measuring methods [10]. The MAE(mean absolute error) in the statistical accuracy measuring method is easy to understand, can visually measure the recommendation quality, is a most widely used measuring method of recommendation quality. This paper capitalizes on the MAE as measuring standard. Through calculating the deviation between predicted user's scores and their actual scores, MAE measures the accuracy: the smaller the MAE, the higher the recommendation quality. There should be three sorts of recommendation quality: global knowledge MAE, interest community-based MAE, Gnutella-based MAE.

The definition of MAE IS shown below:

$$MAE = \frac{\sum_{i=1}^{N} |p_i - q_i|}{N} \qquad (5.2)$$

where N is the number of movies, p_i is the actual scores posed by users on Movie i, q_i is corresponding predicted user scores.

5.4 Experimental Results

Quality Assessment of Interest Community

In the simulated P2P network, we chose different peers to send recommendation inquiries, to formulate variant interest communities. Through observing the similarity between users in the formation of different interest communities every time, we discovered most users who establish interest communities appear in the Top 50 with the biggest interest similarities. However in pure Gnutella networks which don't use the recommendation community-based algorithm, the similarity between randomly generated neighboring peers is quite low. Experimental results demonstrate that the established community with our method boasts a very good quality.

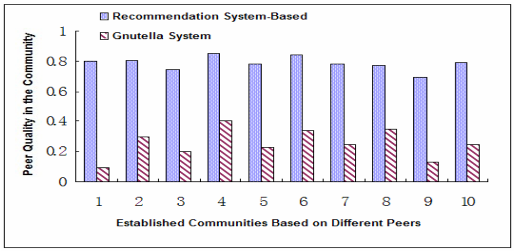


Fig. 5.1: Interest Community Quality Comparison Chart

TTL 's Influence on TOPK Algorithm

Since the searching scope of top-k inquiry is constrained by TTL, when the TTL is increased, the searching scope of top-k inquiry can be expanded, so are the precision rate and the recall ratio as shown in Fig.4. When the TTL is 10, the broadcast-type searching can averagely access 650 peers with the precision ratio 65.2% that of centralized index, and with the recall ratio 70.5% that of the centralized index. In pure P2P environment, the top-k inquiry is a local optimization of searching (peers covered by the query tree), rather than a global one.

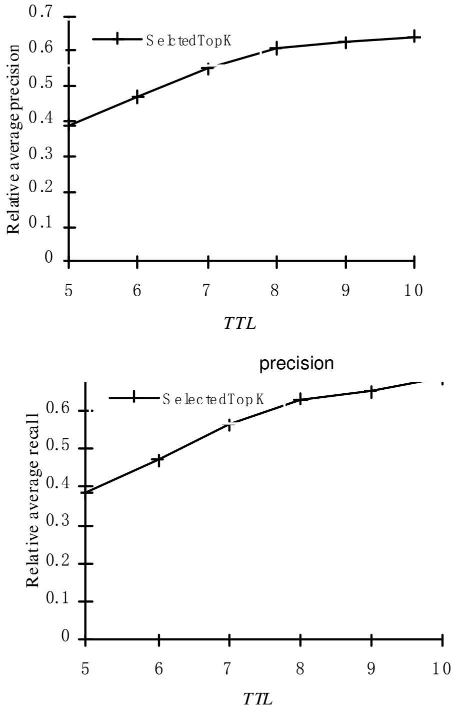


Fig.5.2:Relative average precision and recall of top-k query against different TTL(K=10)

Searching Performance Evaluation

Conduct 30 searches, 10 times for each search, then randomly select one peer to search in the network, ultimately compute the MAE. The result is represented in Fig.5. We compared the interest community-based MAE, Gnutella-based MAE and global knowledge-based MAE. Experimental results indicate that global knowledge-based MAE embodies the smallest error, while Gnutella-based has the biggest error, and interest community-based error is far lower than that of Gnutella.

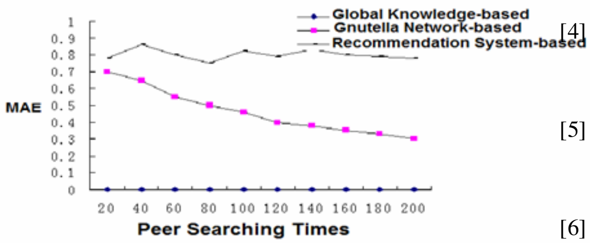


Fig.5.3: MAE against different algorithms

6 Conclusions

Nowadays, the most widely used application of P2P technologies are file sharing systems. These systems extensively apply file mark-based searching techniques to materialize the retrieval of text files. We raised a file recommendation system based on pure peer-to-peer system, which adopts the recommendation help system between users with similar interests to conduct the searching of multimedia non-text files. In such a file sharing system, the interest topological self-adaptive arrangement can be utilized between peers, so as to attain the goal of building a community of peers with similar interests. Experiments illustrate the method has a relatively low absolute mean deviation, efficaciously boosting searching performance. Considering the possibility of peer's fraudulent conducts, further work should be the combination of recommendation system and peer's trust value to hammer out an incentive mechanism in the network, and to guarantee the practicality of the P2P file sharing system

Acknowledgements

This study is supported by National Natural Science Foundation of China（Grant No.60703007）and the Beijing Municipal Education Commission Foundation of China (No. KM201010038001).

References

[1] Huu Tran, Hitchens M, Varadharajan V, Watters P A. Trust Based Access Control Framework for P2P File-Sharing Systems. Proceedings of the 38th Hawaii International Conference on System Sciences-2005.

[2] Palmer Christopher R, Steffan J. Gregory. Generating, Network Topologies That Obey Power Laws.Global Telecommunications Conference, 2000. GLOBECOM '00.IEEE Volume 1, 27 Nov.-1 Dec. 2000, 1(1): 434-438.

[3] Kleinberg J.The small-world phenomenon: An algorithmic perspective. ACM Symp on Theory of Computing, 2000.820828. [C/OL].

[4] Shen, R.-M., et al., "PipeCF: A DHT-based Collaborative Filtering recommendation system," Journal of Zhejiang University: Science, 6 A (2), p. 118-125, 2005.

[5] Wang Yao, Vassileva Julita. Bayesian Network-Based Trust Model. Proceedings of the IEEE/WIC International Conference on Web Intelligence (WI'03).

[6] Magoni Damien.nem: A Software for Network Topology Analysis and Modeling.In:Proc.MASCOTS2002.Los Alamitos, CA: IEEE Computer Society Press, 2002, 364-371. [C/OL].

[7] Palmer CR, Steffan JG. Generating network topologies that obey power law. In: Proc. of the GLOBECOM. San Francisco: IEEE, 2000. 434–438.

Ying Gao received the MS degree in computer science from University of Science and Technology LiaoNing in 2002, and the PhD degree in computer science from the People University of CHINA. Now she is a teacher of Capital University of Economics and Business.Her research interests are in the areas of distributed systems, and database systems.

Jiang Zhan is an Associate Professor in School of Information, Renmin University of China.