Agent-Community based Peer-to-Peer Information Retrieval – an Evaluation

Tsunenori Mine
Faculty of ISEE, Kyushu University
mine@is.kyushu-u.ac.jp

Akihiro Kogo
Graduate School of ISEE, Kyushu University
kogo@al.is.kyushu-u.ac.jp

Makoto Amamiya
Faculty of ISEE, Kyushu University
amamiya@is.kyushu-u.ac.jp

ABSTRACT

The Agent-Community-based Peer-to-Peer Information Retrieval (ACP2P) method uses agent communities to manage and look up information of interest to users. An agent works as a delegate of its user and searches for information that the user wants by communicating with other agents. The communication between agents is carried out in a peer-to-peer computing architecture. The ACP2P is implemented using the Multi-Agent Kodama framework. This paper presents how the ACP2P method works in an agent community network and show the experimental results to illustrate the validity of this approach.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.3.4 [**Information Storage and Retrieval**]: Systems and Software

General Terms

Algorithm, Design, Experimentation, Performance

Keywords

Multi-Agents, Information Retrieval, Peer-to-Peer

1. INTRODUCTION

Peer-to-peer (P2P) computing is an appealing approach not only to file sharing, but also to searching for information relevant to users over the large networks. Recently several studies on making use of document content information have been proposed[1][3][4]. They assume some cooperative environments where gathering all the documents from every peer in the network is possible for classifying all the peers in the network and for creating routing information between peers. However, in uncooperative environments such as business transaction environments, only the information exchanged between peers according to their transaction rules is available to them and such information is acquired only through their transaction process. In the case of Information Retrieval (IR) in such uncooperative environments, a peer can obtain other peer's documents only through their retrieval process. In other words, when a peer issues a query to other peers, the peer can receive only the documents returned by the peers. The retrieved documents a peer acquired can be re-delivered to other peers according to a handling rule of the documents. We are interested in applying agent technologies to such uncooperative environments. Considering the above things, we proposed an Agent-Community-based Peer-to-Peer information retrieval method called ACP2P method, which uses agent communities to manage and look up information related to a user query[6][5]. The agent communities can reflect the structures of human groups or societies such as laboratories, departments, institutions, research groups and so forth, where the people with the same or similar interests, objectives or aims stay together. In the prior research, we have demonstrated the method's effectiveness for reducing communication loads through several experiments[5]. However, we have not so far well discussed the accuracy performance of the method. This paper considers the ACP2P method and discusses the experimental results to illustrate the validity of this approach.

2. OVERVIEW OF THE ACP2P METHOD

The ACP2P method employs three types of agents: user interface (UI) agent, information retrieval (IR) agent and history management (HM) agent. A set of three agents (UI agent, IR agent, HM agent) is assigned to each user. An IR agent communicates with other users' IR agents not only in the community it belongs to, but also in other communities, to search for information relevant to its user's query. A pair of a Query and Retrieved Document Histories (Q/RDH) and a Query and Sender-Agent address Histories (Q/SAH) and retrieved document content files are managed by the HM agent. When receiving a query from a UI agent, an IR agent asks an HM agent to look up target agents with its histories or asks a portal agent to multicast a query to all the IR agents in its community. Where a portal agent is the representative of all IR agents in its community and can multicast a message to them. Any IR agent in a community can ask the portal agent to multicast its query with requesting the number of relevant information to be returned. After multicasting a query, the portal agent only receives the requested number of 'Yes' messages from the IR agents in the order the

AAMAS'06 May 8–12 2006, Hakodate, Hokkaido, Japan.

messages are received and returns them to the query-sender IR agent. When receiving a query from other IR agents, an IR agent looks up the information relevant to the query from its original document and retrieved document content files, sends an answer to the query-sender IR agent, and also sends a pair of the query and the address of the query-sender IR agent to an HM agent so that it can update Q/SAH. The returned answer is either a pair of a 'Yes' message and retrieved documents or a 'No' message indicating that there is no relevant information, although retrieved documents are not returned when the query comes through a portal agent. When receiving answers with a 'Yes' message from other IR agents, the IR agent sends them to a UI agent, and sends them with a pair of a query and the addresses of answer sender IR agents to an HM agent[5].

3. EXPERIMENTS

3.1 Preliminaries

We use the Web pages of Yahoo! JAPAN, which are broadly divided into five categories: animals, sports, computers, medicine, and finance. We select 20 smaller categories from each of them in descending order of the number of Web pages recorded in a category and assign an IR agent to each selected category. A category name is used as the name of an IR agent, and the Web pages in the category are used as the original documents of the agent. To perform the experiments, we compare three methods : (1) ACP2P using the two histories (wQ/SAH for short), (2) ACP2P using the Q/RDH, not using the Q/SAH (woQ/SAH for short), and (3) a simple method always employing a 'multicast' technique (MulCst for short). In the experiments, two query sets : QL=1 and QL=2, are used. QL=1 and QL=2 consist of 10 queries, whose query length is one and two, respectively, where query length means the number of terms in a query.

3.2 Relevance Judgement and Evaluation

In a P2P network environment, gathering all documents from every peer is not always possible, that is, indexing all documents is quite difficult. Thus the first goal for IR in the P2P network environment is to achieve a result comparable with a conventional IR method using a Centralized Indexing DataBase (CIDB method for short) because distributed information retrieval systems, which an IR system in P2P networks belongs to, are not yet better than the "single collection" baseline[3] and the validity of this method is empirically shown[4]; the automatically generated queries and the "single collection" baseline are useful resources in studying federated search (distributed information retrieval) in peer-to-peer networks[4]. As the CIDB method, we employ a probabilistic IR method that applies the BM25 [7] weighting function to all the documents collected from every peer. We use the following equation:

$$\sum_{T \in Q} \log \frac{n+0.5}{N-n+0.5} \frac{2tf}{\frac{dl}{avdl}+tf} \qquad (1)$$

Where Q is a query that contains terms T. tf is the frequency of occurrence of the term within a specific document. N and n are the number of items (documents) in the collection and the number of documents containing the term, respectively. dl and $avdl$ are respectively the document length and average document length, where the document length is the number of terms in a document, and a term is a word detected by a morphological analyzer. In order to compare the ACP2P method with the CIDB method, we used the following equation:

$$\sum_{i=1}^{N_R} \frac{1}{r(i)} / \sum_{i=1}^{N_R} \frac{1}{i}$$

Where $r(i)$ is the CIDB method's rank of the document that is ranked by the ACP2P as the ith document. For example, if a document is ranked by the ACP2P as the 2nd document and the document's rank by the CIDB method is 3, then this means that $r(2)$ returns 3. We call this measure *Reciprocal Rank Similarity* (RRS for short). We can assume that RRS's denominator $\sum_{i=1}^{N_R} \frac{1}{i}$ represents the ideal value of a given model, where it is the CIDB method in this paper. As the ACP2P approaches the given model, the RRS value becomes higher. Thus, the RRS can measure the similarity between ranks generated by the ACP2P and by the CIDB and returns a higher score the smaller $r(i)$ $(1 \leq i \leq N_R)$ is, i.e., the higher the rank. For example, if a user wants to find 3 documents relevant to his/her query and we suppose the top 3 ranked documents' rank returned by his/her agent to be 3, 5 and 1, then the RRS returns $\frac{1/3+1/5+1/1}{1/1+1/2+1/3} = 0.84$, and the top 3 ranked documents' rank to be 3, 5 and 2, then the RRS returns $\frac{1/3+1/5+1/2}{1/1+1/2+1/3} = 0.56$. In the experiment, we use the average RRS: $\frac{1}{N_a} \sum_{i}^{N_a} RRS(i)$, where N_a is the number of all IR agents and $RRS(i)$ is the RRS of the ith IR agent.

3.3 Experimental Results and Discussions

We compare the RRS values of the three methods. All 100 IR agents are assigned to a single community so that we can consider the basic performance of the ACP2P method. The results are shown in Figure 1. In the figure the vertical axis is the average RRS and the horizontal axis is the number of queries issued by each IR agent. As the value of N_R increases, the RRS value also increases and the curve of the graphs becomes flatter. We can see that the RRS value of MulCst increases as the number of queries sent increases. Considering this phenomenon, we surmise that original documents assigned to IR agents will gradually be spread over the community through the document retrieval process of each IR agent. Thus even though a portal agent selects target agents in the order their 'Yes' messages are received, the probability that higher weighted documents will be returned rises. For the same reason, since the RRS value of the MulCst increases as N_R increases, the difference of the three methods decreases. When using QL=1, the wQ/SAH almost achieves higher retrieval accuracy than the other two methods, although the RRS value is unfortunately not so high because the records stored in the content files and the two histories are originally acquired by a portal agent using the query multicasting technique and its RRS value is not so high. We think there are at least two reasons why the RRS score was not so high: 1) a portal agent selects target agents in the order their 'Yes' messages are received, and 2) the ranking results calculated by each IR agent is different from those by the CIDB method even though they use the same equation (1) in Section 3.2 because N used by each IR agent of the ACP2P and that by the CIDB method are different. The first one might be able to be settled by making the portal agent receive the 'Yes'/'No' messages of all IR agents and send them to the query-sender agent although it consumes much more time.

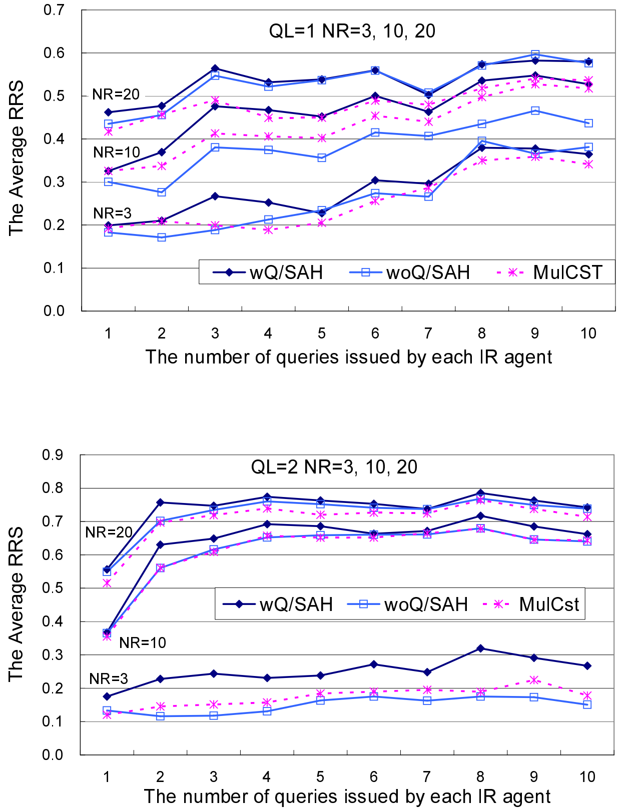

Figure 1: The comparison of average reciprocal rank similarity (RRS) of each IR agent for every query input using 3 different N_R values : N_R=3, N_R=10 and N_R=20. The query belongs to QL=1(Top) and QL=2(Bottom).

However, the second one is not so easy because N used by each IR agent will be changed through its retrieval process. Although there are several methods proposed for distributed information retrieval to estimate the number of documents in each database (DB size for short),[2], [8],[9], they assume such DB size is not varied, thus the number of documents in the network, i.e. N will not be varied either. On the other hand, we assume the DB size will be varied through the retrieval process of IR agents. Therefore a method for estimating N is necessary. In order to estimate N, we first need to estimate the total number of original documents in the network (N_{od}). As a straightforward method, we are trying to use the following simple equation. $N_{od} = \sum_{i \in YES_{IRA}} \frac{N_{QSA_o}}{n_{QSA}} * n_i + N_{QSA_o} * (N_{IRAs} - |YES_{IRA}|)$, where YES_{IRA} is the set of IR agents returning a 'Yes' message o a query issued by a query-sender IR agent. N_{QSA_o} is the number of original documents the query-sender agent has. n_{QSA} is the number of the query-sender agent's original documents relevant to the query. n_i is the number of $IR\ agent_i$'s original documents relevant to the query. N_{IRAs} is the number of IR agents in the network, where 100 in this paper. We assume that N_{IRAs} can be told by the portal agent. $|YES_{IRA}|$ is the number of IR agents in YES_{IRA}. This equation assumes that for every IR agent returning a 'Yes' message, the ratio of the number of original documents relevant to a query and the number of the original documents is the same and the number of the original documents of the IR agent returning a 'No' message is the same as the number of the query-sender agent's original documents. As well as N, we also need to estimate n and $avdl$ in equation (1). n can be estimated by letting the query-sender agent issue a query to all the IR agents returning a 'Yes' message, receive all the documents relevant to the query and count the number of original documents of the IR agents. Where every document should have the information whom it is originally owned or created by. $avdl$ can be set to the average document length of the whole documents the query-sender IR agent has. These are quite rough assumptions, but we have already received positive response from preliminary experimental results, although we need further investigation to show the detail.

4. CONCLUSIONS AND FUTURE WORK

We presented the ACP2P method and discussed the experimental results to illustrate its validity. To do the experiments, we implemented the method with Multi-Agent System Kodama. The experimental results showed that Q/SAH history help to have a higher accuracy in retrieving documents than a method without using the history. Discussing experimental results using more than one hierarchical agent community under more dynamic environments is future work.

Acknowledgment

This research was partly supported by the Grant-in-Aid for Scientific Research (C) (16500082) from the JSPS and the SCOPE-C (052310008) from the MIC, Japan.

5. REFERENCES

[1] M. Bawa, G. S. Manku, and P. Raghavan. Sets: search enhanced by topic segmentation. In *Proc. of SIGIR03*, pages 306 – 313, 2003.

[2] K.-L. Liu, C. Yu, and W. Meng. Discovering the representative of a search engine. In *Proc. of CIKM01*, pages 577–579, 2001.

[3] J. Lu and J. Callan. Content-based retrieval in hybrid peer-to-peer networks. In *Proc. of CIKM03*, pages 199–206, 2003.

[4] J. Lu and J. Callan. Federated search of text-based digital libraries in hierarchical peer-to-peer networks. In *Proc. of ECIR05*, 2005.

[5] T. Mine, D. Matsuno, A. Kogo, and M. Amamiya. Design and implementation of agent community based peer-to-peer information retrieval method. In *Proc. of CIA04, LNAI 3191*, pages 31–46, 9 2004.

[6] T. Mine, D. Matsuno, K. Takaki, and M. Amamiya. Agent community based peer-to-peer information retrieval. In *Proc. of AAMAS04*, pages 1484–1485, 7 2004.

[7] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. Okapi/keenbow at trec-8. In *NIST Special Publication 500-246: TREC-8*, pages 151–162, 1999.

[8] L. Si and J. Callan. Relevant document distribution estimation method for resource selection. In *Proc. of SIGIR03*, pages 298–305, 2002.

[9] J. Teevan, S. T. Dumais, and E. Horvitz. Personalizing search via automated analysis of interests and activities. In *Proc. of SIGIR05*, pages 449–456, 2005.