Cooperative CBR System for Peer Agent Committee Formation

Hager Karoui, Rushed Kanawati, and Laure Petrucci

LIPN, CNRS UMR 7030, Université Paris XIII
99, avenue Jean-Baptiste Clément
F-93430 Villetaneuse, France
{hager.karoui,rushed.kanawati,laure.petrucci}@lipn.univ-paris13.fr

Abstract. This paper deals with the problem of peer agent selection in an unstructured P2P recommendation system. The problem is studied in the context of a collaborative P2P bibliographical data management and recommendation system. In this system, each user is assisted with a personal software agent that helps her/him in managing bibliographical data and recommending new bibliographical references that are known by peer agents. One key issue is to define the set of peer agents that can provide the most relevant recommendations. Here, we treat this problem by using CBR methodology. We aim at enhancing the system overall performances by reducing network load (i.e. number of contacted peers, avoiding redundancy) and enhancing the relevance of computed recommendations by reducing the number of *noisy* recommendations. The peer selection learning cycle is described in detail. Experimental results are also provided and discussed.

1 Introduction

In [7], we have proposed a peer-to-peer (P2P hereafter) collaborative system for bibliographical references management and recommendation. The system, called COBRAS (standing for COoperative Bibliography Recommendation Agent System) aims at: providing help for users to manage their local bibliographical databases and to allow exchanging bibliographical data among like-minded group of users in an *implicit* (i.e. without user request) and *intelligent* (i.e. exchanging relevant data) way. Each user is associated with a personal software agent helping her/him at filling bibliographical records, verifying the correctness of the information entered and more importantly, recommending the user with relevant bibliographical references.

In order to compute relevant recommendations, personal agents collaborate one with each other. A key issue is to define the set of peer agents that can provide the most relevant recommendations. One simple strategy can be to request help from all available agents. However, such a strategy can be expensive or slow if the set of available agents is large, and it is not obvious that it gives the best results in all situations [8]. In this paper, we propose a case-based reasoning (CBR) system for committee recommendation. CBR is a problem solving methodology

S. Joseph et al. (Eds.): AP2PC 2006, LNAI 4461, pp. 51–62, 2008.

[3]. A new problem is solved by finding a similar previous case, and reusing it in the new problem situation. An important feature is that CBR is an approach to incremental, sustained learning since a new experience is retained each time a problem has been solved, making it immediately available for future problems. Our idea is to have a set of interesting peer agents with which the initiator agent will collaborate in a given context. In this system, the initiator agent applies a CBR cycle in order to form a committee. A committee is a set of peer agents supposed to be interesting for a given interest topic. The committee formation is computed when the initiator agent detects some *hot topics* of the associated user. For each detected topic, the agent searchs in his interaction history with other agents in order to choose a subset of peers that are likely to provide relevant references . A CBR-based approach is used for this purpose.

The structure of the paper is as follows. First, we give a global peer-to-peer system overview in section 2. Then, we focus on the committee formation policy in section 3. We give some experimentations in section 4. In section 5, we discuss related work. In section 6, we conclude and we give some directions for future work.

2 System Overview

In COBRAS system, each user is assisted by a personal agent that helps in managing her/his own bibliographical database. Different services are provided by the local assistant such as references edition, references correctness verification and recommendation. We focus on this later service which aims at sharing bibliographic knowledge among the users and taking advantage of past experiences of a single user or even a group of users for recommending more relevant references [7]. Each reference is described by a record containing the following information:

– Bibliographical data: these are the classical data describing a reference such as the type (e.g. Article, In Proceedings, etc), authors, title, etc.
– Keywords: this is a list of keywords describing the reference. The keywords are defined by the user.
– Topics: this is a list of topics the reference is related to. The same topic hierarchy is shared by all users. It has a tree structure and is based on the ACM hierarchy [1] related to the Computer Science domain.

The personal assistant suggests various and interesting recommendations to the associated user according to her/his current activity. The user can either accept or refuse the proposed recommendations. The recommendation computation is made as follows:

– First, the agent applies a simple algorithm described in [7], in order to determine topics from the structure hierarchy that are of current interest to the user. The degree of interest is function of the user activity (i.e. her/his actions on the database).

– For each found topic, the agent sends a recommendation request to a committee of peers. A committee is a set of peer agents that are likely to have references related to the current interesting topic. A recommendation request message is given by: $R = \langle A, T, KL \rangle$ where:
  - A is the sender agent identifier,
  - T is a target topic,
  - KL is a list of keywords that is computed from the set of keywords lists describing references related, directly or indirectly to the topic T.

  A reference is indirectly related to a topic T if it is related to a topic T' more specific than T. In this paper, we focus on this functionnality: the committee formation approach.
– Upon receiving a recommendation request, each agent computes a list of references to recommend.
– The agent merges the received results and proposes the best references to its associated user [7].

3 Committee Formation

The goal of the committee is to enhance the system overall performances by reducing network load and to enhance the relevance of computed recommendations by reducing the number of *noisy* recommendations. The goal consists also to take advantage of the knowledge and experience of other peers. We propose to use a CBR approach in order to exploit history interaction of each agent with others and to learn to find an appropriate committee for each request type. The CBR uses a case base containing relevant evaluated cases. Generally, a case is composed of two parts: the problem part and the solution part (*Case = (Problem, Solution)*). A target problem is a problem to which we search for a solution. It involves a type of recommendation request (in our case, it is the current interesting topic), which presents a part of the user's interests. A case has the following structure: *Case = (T, C)* where:

– $Problem = T$ is a current interesting topic,
– $Solution = C$ is a committee composed of recommended agent to contact according to the topic T.

A CBR cycle is computed for each recommendation request. We describe here the different phases of the CBR cycle for committee formation.

The search phase. Receiving a target problem (a topic T of the computed interesting topic list), the agent selects cases that are similar to the target problem. The committee search is based on a topic similarity which compares the target problem to cases stored in the agent's case base. If the similarity value is above a given threshold σ_t, then the case will be recalled. At the beginning, since the committee case base is empty, the initiator agent sends the recommendation request to all available agents. The topic similarity function is as follows:

$$Sim_{Topics}(T_1, T_2) = 1 - \frac{path(T_1, MSCA(T_1, T_2)) + path(T_2, MSCA(T_1, T_2))}{path(T_1, root) + path(T_2, root)} \quad (1)$$

where:

- $path(a, b)$ returns the path length between nodes a and b,
- $root$ is the topic's tree root,
- $MSCA(a, b)$ returns the most specific common ancestor of nodes a and b in the topic tree.

The same topic map is used by all users. However, we stress that the same hierarchy will be used differently by different users. That's to say the same reference can be related to different topics by different users. For example one may index all CBR-related papers to the same topic, let's say CBR, while another user may index the same papers differently: some related to memory organization in CBR systems and others for CBR case maintenance. A third may index the same references as all related to lazy learning. The topic similarity measure uses the topics underlying hierarchical structure. The applied heuristic is the following: the similarity between two topics depends on the length of the path that links the two topics and on the depth of the topics in the hierarchy. Moreover, a match with specific nodes closer to leaf nodes results in a higher similarity than nodes matching at higher levels of the tree. The heuristic is to return the most specific topics which concentrate a given level of the user's focus.

Reuse Phase. This phase aims at finding a solution to the target problem from a set of source cases found in the previous phase. The solution presents an interesting peer agents committee, to which the recommendation request will be forwarded. The solution committee contains a set of agents computed from the different committees of the source cases found on the previous phase. The *target case = (T, C)*, is such that: T is the initial topic, $C = \cup C_i$, where C_i is the solution of the source case i. The recommendation request will be broadcasted to all peer agents composing the committee C.

Revision Phase. The computed solution is then evaluated by the initiator agent according to the user's evaluation of the recommended references. If the user is interested by a set of recommended references (e.g. the user adds some references to her/his local base). Then, their associated cases and agents will be well evaluated.

Learning Phase. This step consists of adding new cases to the local agent case base. It is the most important step in the CBR cycle. In fact, the selection of retained agents for futur similar problems is done at this stage. As we have explained before, the peer selection is done in a manner to reduce committee size while preserving result quality. The elaboration of a case must be accurate in order to store the relevant information. This phase is based on the agent addition strategy, i.e. the criteria used in order to decide if a given responding agent will be added to the new formed committee or not. A natural idea is to choose all agents which propose some relevant references. Although this simple strategy gives encouraging preliminary results, it does not optimize the committee size. In order to reduce the number of contacted agents, we define criteria which evaluate each agent contribution within the selected committee. We define two criteria-based strategies: heuristics 1 and heuristics 2.

1. **Heuristics 1:** consists of retaining only agents with a local recall value greater than or equal to the average recall value of the references recommending agents. The recall represents the rate of good recommended references among the good existing references ($Recall = \frac{Good\_recommended\_references}{Good\_references}$). Good references are references that are well evaluated by the user. The local recall presents the recall of each agent.
2. **Heuristics 2:** consists of retaining only agents with a local precision value greater than or equal to the average precision value of the recommended references. The precision represents the rate of good recommended references among all the recommended ones ($Precision = \frac{Good\_recommended\_references}{All\_recommended\_references}$). The local precision is the precision of each agent.

4 Experimentation

Experiment settings: n agents which have the same references but they are distributed differently and randomly among the topics of the topic tree. We fix a hot topic, which is considered as a query and we apply our strategy in order to find appropriate agents. We vary each time the number of interesting agents in the system and we compute the recall and the precision. We propose interesting agent term which means agent having good references. In this experiment, we produce the interesting agent as agent having at least x% of the references associated to the current interesting topic. To evaluate our committee formation strategy, we considered three evaluation criteria (recall, precision and committee size). These criteria are of two types :

– Quality criteria: presented by the recall and the precision measures (described in 3).
– Performance criteria: presented in this experiment by the committee size.

The simulation is performed with three different settings:

– *All*: we use a naive approach where the recommendation request is broadcasted to all available agents.
– *Random*: we apply a simple peer selection algorithm, which randomly selects m agents knowing that m corresponds to the number of interesting agents at each time (m varies from 1 to n).
– *Committee*: we apply the CBR-based selection natural approach as described in section 3.

In our experiments, we fixed the number of agent to 10, the used topic similarity threshold σ_t has the value of 0.7. We suppose that an interesting agent is an agent disposing of at least 70% of the reference set associated with the hot topic. A single simulation consists of fixing the minimum number of good references for the interesting agents. Interesting agents do not necessarily have

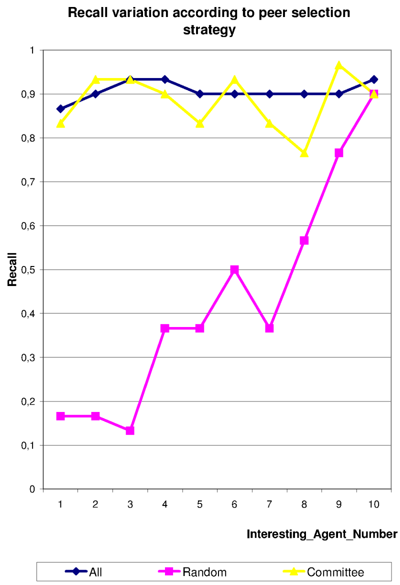

Fig. 1. Recall variation

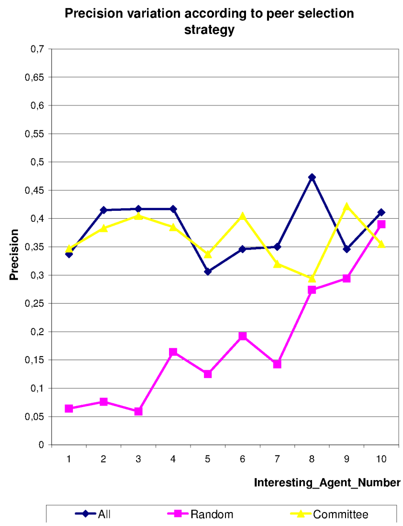

Fig. 2. Precision variation

the same set of good references. The set is chosen randomly. The other references are dispersed among the other topics in a random manner.

Figure 1 shows the recall variation according to the number of interesting agents. We notice that the recall for the *committee* strategy is very close to the *all* strategy and clearly better than the *random* strategy. The recall is often improved by the increase of the number of interesting agents when we randomly choose the agents.

The precision variation is described in figure 2 for the three settings. The *all* and *committee* strategies present more or less similar results, which are better than the naive approach based on random peer selection. However, the precision value is fairly weak with an average of 0.364.

Then, in order to evaluate the performance of the system using the proposed committee formation strategy, figure 3 shows the number of contacted agents among these ten available agents. We notice that the number of contacted agents is reduced. For example in the case of one interesting agent, we solicit 5 agents instead of 10, for 5 and 7 interesting agents, we solicit 8 agents.

Finally, we can say that our natural committee strategy improves the system performance by reducing the number of contacted agents, while it gives similar quality results (i.e. recall and precision) as when all available agents are contacted. However, these results are not satisfactory because we do not want to solicit non interesting agents (without good references), or those which are interesting, but propose the same references as the other agents. In order to improve the results obtained, we studied the effect of applying Heuristics 1 and Heuristics 2 for agents selection (see 3). The results are described in figures 4, 5

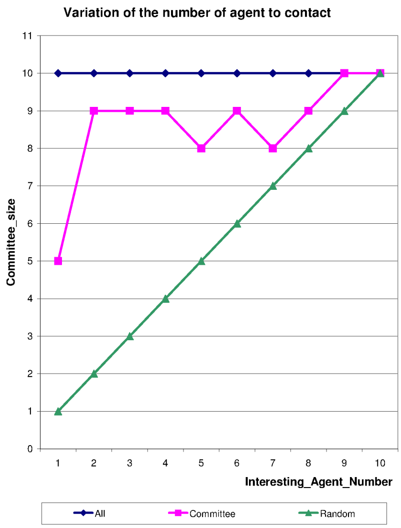

Fig. 3. Committee Size

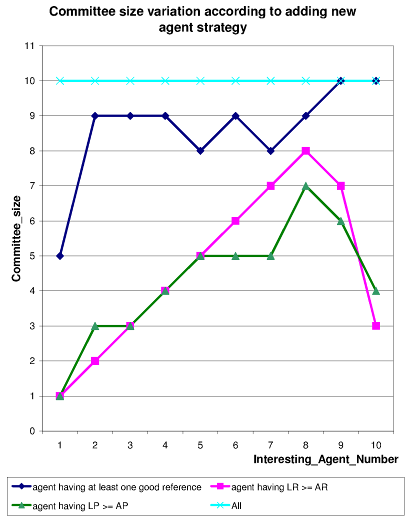

Fig. 4. Committee size variation

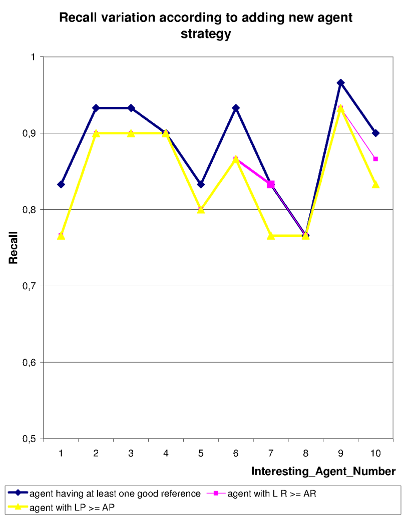

Fig. 5. Recall variation

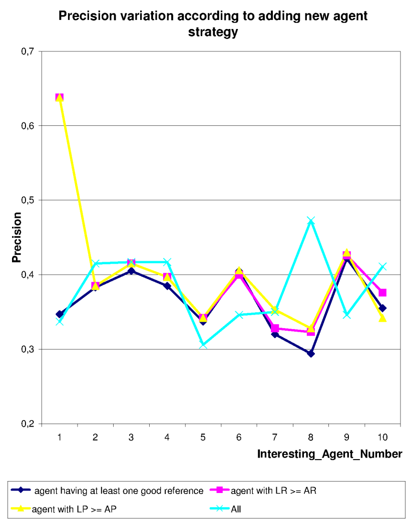

Fig. 6. Precision variation

and 6. Figure 4 shows a clear improvement of the system performance since for both cases (i.e. heuristics 1 and 2), the system solicits at worst all interesting agents. The system contacts even less agents when there is a quite important number of interesting agents. For example, for 6 and 7 interesting agents, the system contacts 6 and 7 agents according to heuristics 1 and respectively 5 and 5 agents according to heuristics 2, compared to 9 and 8 agents for the simple committee strategy (i.e. agent having at least one good reference). The same holds for the 9 and 10 interesting agents, the system solicits respectively 7 and 3 agents according to heuristics 1 and respectively 6 and 4 agents according to heuristics 2, compared to 10 and 10 agents for the simple committee strategy. Heuristics 2 gives, in general, better results than heuristics 1 mainly when there is a quite important number of agents. For example, in the 7 interesting agents case, heuristics 2 retains 5 agents while heuristics 1 retains 7 agents. We conclude that the application of such heuristics gives better system performances. We now examine its impact on the quality criteria (i.e. recall and precision).

Figures 5 and 6 show that the application of the two heuristics gives a recall value similar to the case of contacting all available agents or all agents composing the committee. We also note an improvement of the system precision since we solicit all agents proposing an acceptable contribution (in terms of recall and precision). For example, the precision is improved in the 1, 5, 6 and 9 interesting agents cases. The two heuristics based methods present identical results at the begining, i.e. when the number of interesting agents is lower than 6, and similar results for the other cases. These results show that, even when applying simple heuristics, we succeed in reducing the number of agents to solicit while we keep a very similar result quality, and moreover, we notice an improvement of the precision criterion.

In our experiments, we supposed that an interesting agent is an agent disposing of at least $x\%$ of the reference set associated with the hot topic. We varied the x and we studied its effect on the committee formation evaluation criteria. The experimental results are described in figures 7, 8 and 9. These results are obtained by adding heuristics 2 to the simple committee formation strategy. We note that, for the different values of x, the curves have the same trend. We remark

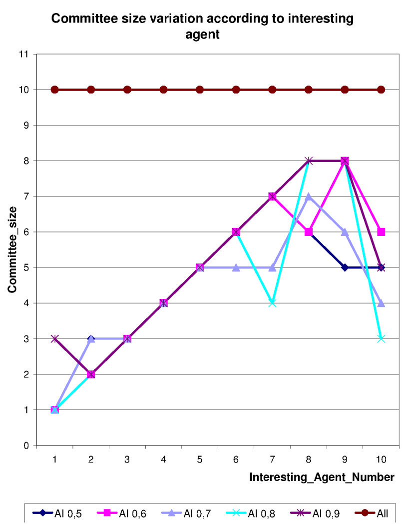

Fig. 7. Committee size variation

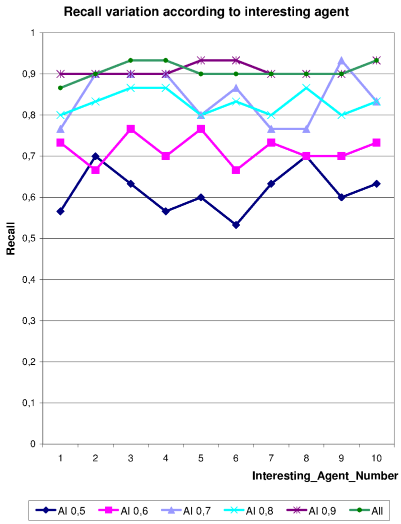

Fig. 8. Recall variation

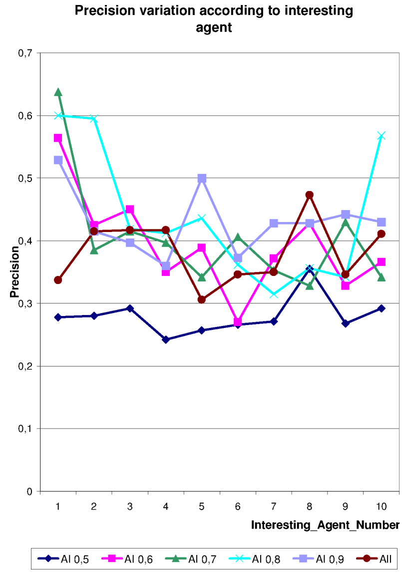

Fig. 9. Precision variation

also that, in all cases, the number of retained agents is reduced while maintaining similar result quality (i.e. recall and precision) or even an improvement. Although the results obtained are acceptable and encouraging, we think that the results (mainly the precision which is quite low) presented will be improved by introducing some constraints in the committee formation process such as:

– using better reference similarity taking into account semantic criteria (e.g. same authors, same conference, etc). This will improve the quality of recommendation and the precision of the system.
– handling the redundancy problem between agents results. In many cases, some of the references proposed by interesting agents are the same. So it is useful to verify this before contacting all possibly interesting agents.
– proposing an appropriate cooperative approach to up to date committee according to the changing user's interests in a dynamic network. This constitues our present work.

5 Related Work

Different committee formation approaches are proposed in the literature. Some are based on the notion of agent reputation [5] or agent expertise [4]. Others propose to apply automatic learning techniques in order to enable each agent to determine if it needs to increase the committee of peers and, if it is the case, which peer agent to invite [8]. For our purposes, the idea consists of providing each peer agent with the capacity of selecting a subset of peer agents having good results according to a given recommendation request type (in our case, the recommendation of bibliographical references). The goal is to improve the performance of the whole system by reducing the network and the agents charge.

– Bibster system (standing for Semantic-Based Bibliographic Peer-to-Peer System)[4], has a peer-to-peer architecture and aims at sharing bibliographic data between researchers. The peer selection is based on the *expertise* notion [6]. The expertise is a set of ACM topics. All system peers share a common ontology for publishing semantic descriptions of their expertise in a peer-to-peer network. This knowledge about the other peers expertise forms the semantic topology, which is independent of the underlying network topology. When a peer receives a request, it decides to forward the query to peers whose expertise is similar to the subject of the query. Peers decide autonomously to whom advertisements should be sent and which advertisements to accept. This decision is based on the semantic similarity between expertise descriptions. This strategy gives good results compared to broadcasting the query to all or to a random set of peers but does not exploit past experience to learn and improve the formed semantic topology.
– Gupta et. al. [5] propose a reputation system for decentralized unstructured P2P networks like Gnutella [2] for searching and information sharing. The peer selection strategy is based on the agent *reputation* notion. The reputation system uses objective criteria to track each peer's contribution in the system and allows peers to store their reputations locally. They propose two alternate computation mechanisms for a reputation system that objectively map each peer's activity in the P2P network to a dynamically updated reputation score. The two mechanisms are the debit-credit reputation computation (DCRC) and the credit-only reputation computation (CORC). The first mechanism (DCRC), credits peer reputation scores for serving content and debits for downloading. The second one (CORC), credits peer reputation scores for serving content but offers no debits. The expiration on the scores instead serves as a debit. A reputation score is intended to give a general idea of the peer's level of participation in the system. Reputation scores are based on two essential factors: the peer capability and its behavior. The capability of a peer depends on its processing capacity, memory, storage capacity, and bandwith. The behavior of a peer is determined by the level of contribution offered by it for the common good of the P2P network. Peers are free to enroll in the reputation computation or not. A reputation computation agent (RCA) is used for enrolling peers who wish to enroll in reputation computations and for updating peer reputations in a secure, light-weight, and partially distributed manner. Having reliable reputation information about peers can form the basis of an incentive system and can guide peers in taking decisions.
– Ontañón and Plaza [8] propose another strategy of selection of the agents that join a committee for solving a problem in the classification tasks. The basic reason of the incentive of agents to cooperate in the form of a committee is that they can improve their performmance in solving problems. The

committee organization improves (in general) the classification accuracy with respect to individual agents. It is a learning framework that unifies both the when and the who issues. In fact, the agent learns to assess the likelihood that the current committee will provide a correct solution. If the likelihood is not high, the agent has to invite a new agent and has to decide which agent to invite. The agent learns to form a committee in a dynamic way and to take decisions such as whether it is better to invite a new member to join a committee, when to individually solve a problem, when it is better to convene a committee.

We have chosen to propose a new strategy of committee formation which will be dynamic, extensible and adaptable. The proposed strategy exploits as much as possible past experiences and will be adaptable with the new real constraints. To ensure this, our strategy relies on a case-based reasoning system. It aims at computing committee's recommendations. In fact, when an agent detects a hot topic, it applies a CBR cycle to find some committee recommendation associated with the request type. The reference recommendation request will then be forwarded to peer agents composing the recommended committee.

6 Conclusion

We have presented a cooperative CBR approach for peer committee recommendation in a bibliographical references recommendation system COBRAS. The agents cooperate with each other in order to share their knowledge and their past experience to improve their efficiency.

We proposed a strategy allowing an agent to determine peer agents committee for a given recommendation request. This strategy uses a CBR technique in a cooperative way allowing for reusing and sharing of knowledge and experience.

The results obtained are encouraging. Different tracks however should be explored in order to improve both the quality and the performance criteria: handling the problem of agent redundancy in a commitee; proposing a strategy to maintain the agent case base and ensuring up-to-dated committee according to user's interest changing. These perspectives are the subject of our present work.

References

1. Acm, `http://www.acm.org/`
2. Gnutella, `http://gnutella.wego.com/`
3. Aamodt, A., Plaza, E.: Case-based reasoning: Foundational issues, methodological variations, and system approaches. AI Communications 7(1), 39–59 (1994)
4. Broekstra, J., Ehrig, M., Haase, P., Harmelen, F., Menken, M., Mika, P., Schnizler, B., Siebes, R.: Bibster -a semantics-based bibliographic peer-to-peer system. In: Proceedings of SemPGRID 2004, 2nd Workshop on Semantics in Peer-to-Peer and Grid Computing, New York, USA, May 2004, pp. 3–22 (2004)
5. Gupta, M., Judge, P., Ammar, M.: A reputation system for peer-to-peer networks. In: Proceedings of ACM Networks and Operating System Support for Digital And Video NOSSDAV 2003, Monterey, CA (2003)

6. Haase, P., Siebes, R., Harmelen, F.: Peer selection in peer-to-peer networks with semantic topologies. In: International Conference on Semantics of a Networked World: Semantics for Grid Databases, Paris (2004)
7. Karoui, H.: Agent RàPC pour la gestion coopérative de bases bibliographiques personnelles. In: Plate-forme AFIA 2005, 13 ème atelier de Raisonnement à Partir de Cas, Nice, France (May 2005)
8. Ontanón, S., Plaza, E.: Learning to form dynamic committees. In: Proceedings of the second International joint Conference on Autonomous Agents and Multiagent Systems, Melbourne, Australia, pp. 504–511. ACM Press, New York, USA (2003)