Distributed Semantic Search for the Web: A Multiagent Approach

(Extended Abstract)

Murat Şensoy
Department of Computing Science
University of Aberdeen
AB24 3UE, United Kingdom
m.sensoy@abdn.ac.uk

ABSTRACT

In this paper, we propose to combine Semantic Web technologies and multiagent systems in a novel way to enable users to locate and share URLs relevant to their search interests.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent Systems

General Terms

Design, Experimentation

Keywords

Semantic Web, Search, Social Networks

1. INTRODUCTION

In this paper, we propose an agent-based approach for collaborative semantic search. Our approach does not depend on the semantically annotated Web resources; instead it depends on reusing the search results refine by users. In our approach, each user is represented by an agent, which has a local ontology to maintain useful search concepts. A search concept corresponds to a semantically described search interest. For the representation of search concepts, Description Logic (DL) and OWL [2] are used. In this setting, searching the Web for a specifi interest is formulated as findin instances of the related search concept. When a user needs to search the Web for specifi information or interest, he describes a new search concept and his agent interacts with other agents to retrieve related search results.

2. COLLABORATIVE SEARCH

Current Semantic Web technologies enable description of complex concepts using DL and makes reasoning on these concepts possible and feasible. Therefore, instead of using ambiguous keyword-based queries, we propose to use Semantic Web technologies to explicitly and clearly describe search interest of users. That is, each search interest is represented as a *search concept* using an OWL ontology and DL as shown in Example 1.

Cite as: Distributed Semantic Search for the Web: A Multiagent Approach (Extended Abstract), Murat Şensoy, *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, van der Hoek, Kaminka, Lespérance, Luck and Sen (eds.), May, 10–14, 2010, Toronto, Canada, pp. 1561-1562

EXAMPLE 1. *The search interest "Comedy Videos that contain a muscular guy who is chased by a small dog" can be represented as* $Comedy \sqcap Video \sqcap \exists contain.(Small \sqcap Dog \sqcap \exists chase.(Muscular \sqcap Man))$. *On the other hand, the search interest "Comedy videos that contain a muscular dog which is chased by a small man" can be represented as* $Comedy \sqcap Video \sqcap \exists contain.(Small \sqcap Man \sqcap \exists chase.(Muscular \sqcap Dog))$.

In this paper, we envision a multiagent system that consists of agents representing human users. Each agent has access to a common meta-ontology that contains primitive concepts and properties (e.g., concepts like Man, Dog, Video and so on). This ontology is static and does not contain any search concept. This ontology is public; i.e., may be downloaded from a well-define resource. It constitutes grounding for describing search concepts and sharing this description between the agents. In addition to the common ontology, each agent has a local ontology. This ontology contains the search concepts known by the agent. Each search concept is described using only the concepts and the relations from the common ontology. When an agent makes descriptions of its search concepts available to others, other agents can interpret and reason about these search concepts easily using a DL reasoner, such as Pellet [1]. Each search concept is define using the concepts from the common ontology. Hence, given a set of search concepts, an agent can reason about the relationships between these concepts using a DL reasoner (e.g., subsumption hierarchy).

When a user has a search interest, he interacts with his agent and describes his search interest using an interface. The agent converts the described search interest into a DL representation. Using this representation and a DL reasoner, the agent searches its local ontology for a semantically equivalent search concept. If an equivalent search concept is found, instances of the found concept are retrieved from the ontology. Each retrieved instance corresponds to a URL of a Web resource. Retrieved instances are returned to the user. If an equivalent concept is not found or instances of the found search concept are not enough, the agent interacts with other agents to locate Web resources related to the search interest of its user. These agents are called neighbors and determined as follows. First, the agent determines the most similar search concepts to the new search concept using ontological reasoning and a predefine similarity metric [3]. Then, the agents that have used these concepts in their previous interactions are regarded as neighbors. Note that, during its interaction with others, an agent records which concepts are used by which agents.

An agent creates a new search concept when its user has a new search interest and none of the known search concepts in its local ontology can completely describe the interest. We may note that the knowledge of the agent about existing search concepts in the society is limited. This means that the new concept may either already exist in the society but the agent may not be aware of it or the concept may be totally new to the entire society. In order to differentiate between these two cases, the agent queries its neighbors to fin out if the search concept is already known to its neighbors.

If none of the neighbors locates the desired search concept within their ontologies, the agent concludes that this search concept is not known by any of its neighbors. In this case, the agent places the concept into its local ontology with a unique name and informs its neighbors; otherwise the agent names the new concept using the names learned from its neighbors. Therefore, the agent and the neighbors can understand each other during their future communications regarding this search concept. Note that, in many cases (i.e., especially during bootstraping), the agent may need to expand its knowledge about the society by discovering new agents with a specifi search interest. For this purpose, various P2P mechanisms or directory services can be used.

In the proposed approach, when an agent generates a new search concept to represent its new search interest, it teaches this search concept to its neighbors by sharing the description of the concept or the neighbors inform the agent about the search concept if the concept is already known by them. This leads to an interactive learning of new search concepts. Hence, mutually understood search concepts emerge as a result of agents' social interactions.

For a given search interest, the agent can retrieve search results as follows. First, using a reasoner, the agent queries its local ontology for a search concept corresponds to its current search interest. If a semantically equivalent concept C is found, the agent retrieves direct and indirect instances of C from its local ontology. On the other hand, if none of the know search concepts corresponds to search interest, a new search concept is created and added into its ontology as explained above. During the creation of new concept, the agent does not only learn related search concepts from its neighbors, but also discovers other peers that have the same or similar search interests. If the agent currently does not know enough number of Web resources related to the search interest, in order to learn other instances of the search concept, the agent queries its neighbors, which may be knowledgeable about the search concept. When queries, a neighbor tries to locate the related search concept C and C's sub-concepts in its local ontology. Then, the neighbor rank the instances of these concepts and sends the best n_{max} instances and their type to the sender. We may note that all instances of C's sub-concepts are also valid instances of C. Therefore, even though the neighbor does not have C in its local ontology, it may send the instances of C's sub-concepts that exist in its ontology.

Given an agent with a specifi search interest, the proposed approach depends on the assumption that users of some agents in the society have already searched the Web for the same or similar search interests. Hence, if the agent can locate those agents, it can retrieve URLs of the most related Web resources without bothering its user with mining the messy results returned by the search engines. However, in many settings, the retrieved results from other agents may not be enough (e.g., during bootstrapping). If this is the case, the user may be prompted to search the Web using conventional tools such as search engines. Moreover, in order to enable better exploration of the Web, the agent encourages its user to use those tools even if enough number of the Web resources about the search interest is already known. As the user discovers new Web resources for his search interest, the agent is informed so that the agent can update its local ontology with the discovered resources and share this information with others in the future as proposed in this paper.

3. EVALUATION

In order to evaluate our approach, we created a meta-ontology[1] and conducted experiments using 100 search agents. Each agent is randomly given an initial acquaintance so that the underlying network of agents is connected. Also, we have created 10 distinct roles, each of which is composed of 10 semantically related search interests. In our experiments, each agent plays one role and changes its role with a small probability 0.05. Using these settings, we evaluate our approach in terms of emerging search concepts, messaging overhead, search results, and network topology.

During our experiments, agents have created 160 search concepts cooperatively. However, an agent knows only 12% of these concepts and only 20% of them are created by the agent itself. We have examined the emergent topology of the agent network and observed that the interactions between agents lead to highly connected virtual communities (clusters), in which agents have similar search interests (i.e., play the same role). The resulting communities represent different search interest themes; that is, useful search concepts created in a community may be useless in another one. Therefore, agents tend to add new search concepts created in their community to their ontology, instead of learning the search concepts created in other communities. Interestingly, even though the ontologies of agents in the system are becoming highly different as new search concepts are created, ontologies of the agents in the same community are becoming more similar over time. It seems that, as in real life, agents in each virtual community create their own jargon to efficientl communicate regarding their common search interests.

We have observed that only a small amount of URLs relevant to a search interest are learned by an agent from directly its user. This number is 4.0 in the beginning of the experiments, and rapidly decreases to 1.2 after a while. On the other hand, average number of relevant URLs learned from other agents increases from 0.5 to 21. This means that, agents successfully discover others with similar search interests over time and make use of the URLs previously refine by the users having same or similar search interests.

4. CONCLUSIONS

In this paper, we propose an agent-based approach for distributed semantic search on the Web. Our approach enables human users to describe their search interest to their agents. Our analyses of the proposed approach show that depending on the interests of their users, agents evolve their ontologies and create new search concepts. Based on these search concepts, the agents coordinate and compose virtual communities. In open environments like the Web, some of the agents may be malicious and distribute bogus information. Furthermore, some search concepts may be subjective (e.g., funny videos) and may be interpreted differently by different users (i.e., a video may be funny for a user, while it is not for another one). In order to extend our approach, we plan to integrate techniques for agents to compute trustworthiness of their peers.

5. REFERENCES

[1] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, and Y. Katz. Pellet: A practical OWL-DL reasoner. *Web Semant.*, 5(2):51–53, 2007.

[2] M. K. Smith, C. Welty, and D. L. McGuinness. OWL: Web ontology language guide.

[3] A. Tversky. Features of similarity. *Psychological Review*, 84(4):327–352, 1977.

[1] Available at http://www.csd.abdn.ac.uk/~murat/websearch.owl.