An Agent-Oriented Personalized Web Searching System

Tarek Helmy Satoshi Amamiya Tsunenori Mine Makoto Amamiya

Department of Intelligent Systems
Kyushu University
6-1 Kasugakoen, Kasuga-shi
Fukuoka 816-8580, Japan
Tel: 81-92-583-7615, Fax: 81-92-583-1338
[helmy,roger,mine,amamiya]@al.is.kyushu-u.ac.jp

Abstract. Web retrieval is now one of the most important issues in computer science, and we believe that applying multi-agent systems to this area is a promising approach. We introduce Kodama[1] system, which is being developed and in use at Kyushu University, as a multi-agent-based approach to build a distributed Information Retrieval (IR) system that lets users retrieve relevant distributed information from the Web. We reported methods to agentify the Web, and to cluster the agentified domain into communities. In order to investigate the performance of our system, we carried out several experiments in multiple Server Agent domains and developed a smart query routing mechanism for routing the user's query. The results ensure that the idea of Web page agentification, clustering and routing techniques promise to achieve more relevant information.

1. Introduction

With the exponentially growing amount of information available on the Internet, the task of retrieving relevant information consistent with the user's information need has become increasingly difficult and the users normally face with very large hit lists with low precision. The information gathering and retrieving processes in the traditional search engine are independent of the User's Preference (UP), and therefore feedback from the later process is hardly adaptive to improve the quality of the former process. Kodama project starts in response to the need for a new kind of agent-oriented IR system that is completely different from the traditional search engines populated on the Internet. Researchers in Artificial Intelligence (AI) and Information Retrieval (IR) fields have already succeeded in developing multi-agent based techniques to automate the management of information flooding [1]. A way to partially address the scalability problems posted by the size and dynamic nature of the Web is to divide the Web into localized Sever Agents (SA) that agentify specific domains by a set of Web Page Agents (WPA) developed in our project [2,3,4]. We will start by describing the agentification of the Web servers and discuss the methodologies of clustering the agents into communities. We introduce the routing mechanism of Kodama and the evaluation of Kodama agents.

2. Web Server Agentifcation

Cooperating intelligent Kodama agents are employed to agentify the Web servers where the infrastructure is preexisting in the form of Web links. Kodama system uses three types of agents in the agentification mechanism for searching the Web. A SA assigned to each Web server, a WPA assigned to each Web page, and a User Interface Agent (UIA) assigned to each user's machine [2,3,4]. The SA starts from the portal address of the Web server and creates the hyper structure of WPAs communities based on the hyper link structure in the Web server, (see Figure 1). The SA

[1] Kyushu University Open Distributed Autonomous Multi-Agent

knows all the WPAs in the server and works as a gateway when the WPAs communicate with each other or with one in another server. The SA clusters the WPAs into communities and automatically defines its attributes to be used in the routing mechanism. The WPA analyzes and continually keeps track of the content of its Web page. Each WPA has its own parser, to which the WPA passes a URL, and an IP vector, in which the WPA keeps all the policy keywords found in its URL. The WPA takes essential properties and principles given by the SA to create the IP as ontology that represents the context of the Web page. The created WPAs register themselves to the SA and write all the words into an IP file. An SA of *n* Web pages creates one IP file in which the terms of the Web pages are represented as *n* vectors of keywords with a weight value assigned to each keyword and modified according to the user's responses ($\Re$). The WPA uses the IP to decide whether or not the user's query (q_i) belongs to the WPA. At the retrieval phase, when the WPAs receive a q_i from a SA, initiate search by interpreting the q_i and/or either asking "Is this yours?" or announcing "This is yours," to its down-chain WPAs. The selected WPAs and/or their down-chain WPAs of each Web server, which in turn, interpret the q_i according to their IPs and reply the answer "This is mine" or "Not mine" with some confidence. The UIA is designed to learn the UP either implicitly or explicitly from his/her browsing history. We have developed Kodama's browser and investigated some sensors in correlation with the time of visiting the page to let the UIA detects autonomously the actual user's implicit response. The UIA resides in the user's machine, communicates with the WPAs via a SA to retrieve information relevant to the q_i, and shows the results returned by the WPAs to the user after filtering and re-ranking them. The UIAs in Kodama system look over the shoulders of the users, receive $\Re$ of his/her interest/not interest to the results and regard them as rewards to adapt the UP files. The UIA uses UP to predict a user's action based on the similarity of the current query to the already learned UP.

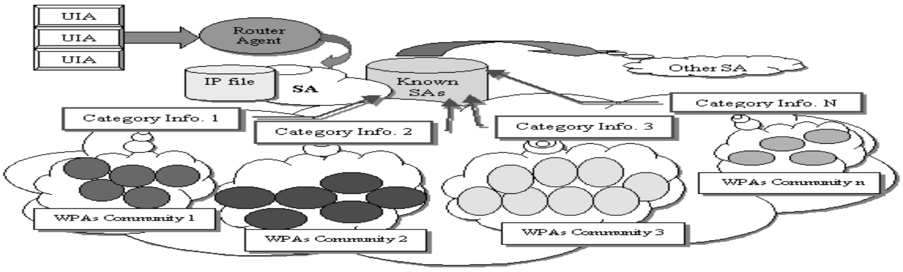

Fig. 1. The hierarchical structure of an agentified domain

3. Creating the Agents Communities

While agentifying the Web server and creating a WPA for each Web page based on the pre-existing hyper link structures, the SA will add to its known SAs table all the portal addresses of the external links, which point to other portals. This means that, the SAs community will be created automatically while the agentification process. For instances, while agentifying the Web server of Kyushu University http://www.kyushu-u.ac.jp/, if there is an external link, i.e. http://www.osaka-u.ac.jp/, in one of the Web pages of Kyushu University's Web server, then this portal; http://www.osaka-u.ac.jp/; address will be added as a community member of the SA of Kyushu University. This means, the WPAs will send "friend of mine" messages to the SA to register the portal addresses of the external links in their Web pages as a relevant one. The SA will cluster the WPAs into communities based on the incoming keywords of the q_i over time. The server's administrator may define keywords as seeds for clustering the Web pages of the server. We introduce a definition of WPAs community that enables the SA to effectively focus on related subsets of WPAs to increase the precision of search results. The name of a cluster is initially constructed from the q_i and is dealt with the main attribute of the cluster. The cluster's name is updated according to newly inputted queries related to the cluster and a set of keywords surrounded by specific HTML tags that are included in the cluster's Web pages relevant to the queries. This means that, over time the communities of WPAs will be refined so that an agent may be assigned to or released from specific community. Following, we show the definition to create a cluster of WPAs. Let Q be a set of cluster names $\{CNq_i \mid 1 \le i \le n, CNq_i = \{w_j \mid 1 \le j \le m\}\}$, where w_j is a keyword, n is the number of elements in Q. We call the number of elements in a set, size. Thus, n is a size of Q, and m is a size of $CNqi$. Let q be a user's query, such that $q = \{w_j \mid 1 \le j \le l\}$ (l is a size of q) and ϕ be an empty set. The clustering procedure is as following. $Q \leftarrow \phi$, enter a query q_i. If $q_i \cap CNq_j = \phi$ for any $CNq_j \in Q$, then create new cluster Cq_i that consists of a set of Web pages relevant to q_i. Then, q_i is assigned to CNq_i, which is the name of Cq_i, i.e. $CNq_i \leftarrow q_i$ and $Q \leftarrow Q \cup CNqi$. For each $CNq_j \in$ Q, if $q_i \cap CNq_j \ne \phi$ and $q_i \not\subset CNq_j$, then $Q \leftarrow Q \cup CNqi$, $CNq_i \leftarrow CNq_i \cup q_i$ and $CNq_i \leftarrow CNq_i \cup k_j$ for every $k_j \in Tag$. Where Tag is a set of keywords surrounded by specific tags in such Web pages that are in Cq_j and relevant to q_i.

4. Routing Mechanism of Kodama

Although a single router is scalable enough to potentially handles of thousands of SAs. In practice it is desirable to run a separate router for relevant SAs of a common topic. For instances, the SAs of AAAI, IEEE and ACM portals belong to one router agent. The router delegates the given query to the most popular and relevant SA. For each community of SAs, there is a router agent that holds a set of attributes as ontology of each SA (see Table 1). Where A_{11} to A_{1m} means the set of attributes automatically determined by clustering the WPAs and adapted by the system to reflect the ontology of the SA_1. *W* is the weight value, which assigned to each attribute and continually adapted based on the $\Re$ from the UIAs. Routing refers to the process of selecting the relevant SA and forwarding queries to it to retrieve the Web pages consistent with user's information need. Relevancy is used to determine the popularity of the SA for a particular type of queries. Kodama system maintains the similarity S_j^i between q_i and the attribute fields of SA_j of the known SAs's table using the following formula. $S^i_j = \sum_i w_{j,i} \cdot g(k_i)$. Where, $g(k_i) = 1$ if $k_i \in q_i \cap SA_j$, otherwise $g(k_i) = 0$.

Table 1. Server Agent's Atrributes

Relevant SA	Attribute Information of a SA			
SA_t	A_{11},W_{11}	A_{12},W_{12}	…	A_{1m},W_{1m}
SA_2	A_{21},W_{21}	A_{22},W_{22}	…	A_{2m},W_{2m}
SA_j	A_{31},W_{31}	A_{32},W_{32}	…	A_{3m},W_{3m}
SA_x	A_{x1},W_{x1}	A_{x2},W_{x2}	…	A_{xm},W_{xm}
SA_n	A_{n1},W_{n1}	A_{n2},W_{n2}	…	A_{nm},W_{nm}

5. Distributing Web Search on Kodama

There should be a single entity that controls the list of router agents. When you register a router agent, it goes through one of several dozen routers who work with Kodama to add names to the list. Kodama, in turn, keeps a central database known as the router's database that contains information about the profile of each router. Each of the routers has thousands of SAs and handles its requests. While registering the SA into a router agent, the SA sends the names of its clusters to be used for routing relevant queries into that SA. The routers are specialized agents that send your query and those of every other UIAs to their relevant SAs along thousands of pathways of SAs. When the router receives a query from the UIA, the router does the followings with it. It asks for a list of relevant SAs. If the router agent found relevant SAs, it assigns the request with specific SA because it already knows that this SA is relevant to this query. Then, it merges the results of the query and sends them back to the UIA. Otherwise, it forwards the queries to other routers if the results do not satisfy the user or the router could not find a relevant SA. It may return an error message because the router could not find any relevant SA.

6. Experimental Results

We have performed several experiments to make a consistent evaluation of Kodama system performance. In the experiments, we agantified fifty Web servers by giving the portal addresses of the Web servers to the system; the number of Web pages within the agentified servers varies from 300 to 2500 pages. The system creates the hyper structure of the WPA communities based on the hyperlink structure of each Web server and creates the SA's attributes to be used in the router side for the routing. Then, the UIAs sent some queries to the system. We calculated the Precision of the retrieved URLs to user's queries. The results (see Fig. 2) show that the idea of Web page agentification and the routing mechanisms promise to achieve more relevant information to the users and also promoted using Kodama as a PinPoint IR system.

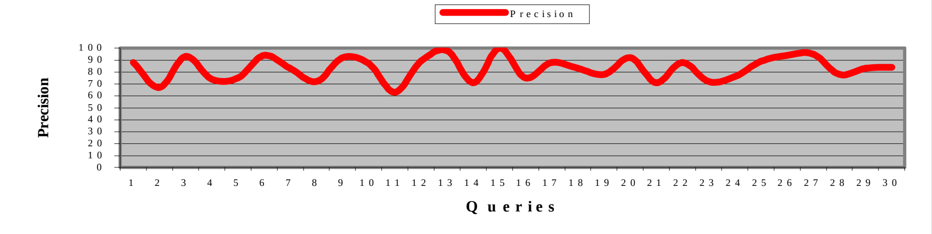

Fig. 2. Precision of the queries submitted to the system

7. Conclusion and Future Work

We introduced methods to agentify the Web, and to cluster the SAs and WPAs into communities. We also introduced the routing mechanism of Kodama system to select the most relevant SA to the given query. We carried out experiments to investigate the performance of Kodama system. Through these experiments, we ensure that Kodama's techniques promise to achieve more relevant information to the users. Currently, the routing of Kodama is a simple query routing that binds to two hierarchical levels of router agents. We plan to scale it by increasing the number of SAs and developing more sophisticated routing mechanism for maintaining multiple hierarchy of router agents.

8. REFERENCES

[1] Edmund S. Yu, Ping C. Koo, and Elizabth D. Liddy, "Evolving Intelligent Text-based Agents", Proc. of the 4th International Conference of Autonomous Agents, June 3-7- 2000, Spain, pp.388-395.

[2] T. Helmy, S. Amamiya and M. Amamiya, "Collaborative Kodama Agents with Automated Learning and Adapting for Personalized Web Searching", Proc. of the 13th Inter. Conference on Innovative Applications of AI (IAAI/IJCAI-2001), pp. 65-72, August 7-9, 2001, USA.

[3] T. Helmy, S. Amamiya and M. Amamiya, "Pinpoint Web Searching and User Modeling on the Collaborative Kodama Agents", LNCS Proc. of the 2nd Inter. Conference on Electronic Commerce and Web Technologies EC-WEB 2001, pp. 305-314, Sept. 2001, Germany.

[4] T. Helmy, S. Amamiya, and M. Amamiya, "User's Ontology-Based Autonomous Interface Agents", The Second Inter. Conference on Intelligent Agent Technology (IAT2001) Proc. book entitled "intelligent Agent Technology: Research and Development", pp. 264-273, October 23-26, Japan.