ELSEVIER

Decision Support Systems 35 (2003) 167–183

Decision Support Systems

www.elsevier.com/locate/dsw

Design and evaluation of a multi-agent collaborative Web mining system

Michael Chau*, Daniel Zeng, Hsinchun Chen, Michael Huang, David Hendriawan

Department of Management Information Systems, Eller College of Business and Public Administration, The University of Arizona, Tucson, AZ 85721, USA

Abstract

Most existing Web search tools work only with individual users and do not help a user benefit from previous search experiences of others. In this paper, we present the *Collaborative Spider*, a multi-agent system designed to provide post-retrieval analysis and enable across-user collaboration in Web search and mining. This system allows the user to annotate search sessions and share them with other users. We also report a user study designed to evaluate the effectiveness of this system. Our experimental findings show that subjects' search performance was degraded, compared to individual search scenarios in which users had no access to previous searches, when they had access to a limited number (e.g., 1 or 2) of earlier search sessions done by other users. However, search performance improved significantly when subjects had access to more search sessions. This indicates that gain from collaboration through collaborative Web searching and analysis does not outweigh the overhead of browsing and comprehending other users' past searches until a certain number of shared sessions have been reached. In this paper, we also catalog and analyze several different types of user collaboration behavior observed in the context of Web mining.

Keywords: Web searching; Web content mining; Collaborative information retrieval; Collaboration behavior; Collaborative filtering; Multi-agent systems; Software agents; Post-retrieval analysis

1. Introduction

It has become increasingly difficult to search for useful information on the Web, due to its large size and unstructured nature. Researchers have developed many different techniques to address this challenging problem of locating relevant Web information efficiently. Examples of such techniques include Web search engines, meta-searching, post-retrieval analysis, and enhanced Web collection visualization [10,18,36,44,48].

A major problem with most such techniques is that they do not facilitate user collaboration, which has potential for greatly improving Web search quality and efficiency. Without collaboration, users must start from scratch every time they perform a search task, even if other users have done similar or relevant searches.

In this paper, we propose a multi-agent approach for collaborative information retrieval and Web mining, implemented in a system called the *Collaborative*

[*] Corresponding author. Tel.: +1-520-626-9239.
E-mail addresses: mchau@bpa.arizona.edu (M. Chau), zeng@bpa.arizona.edu (D. Zeng), hchen@bpa.arizona.edu (H. Chen), mhuang@bpa.arizona.edu (M. Huang), dhendriawan@yahoo.com (D. Hendriawan).

0167-9236/02/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.
PII: S0167-9236(02)00103-3

Spider. The main research issues explored include (a) the impact on users' search and analysis performance and the optimal volume of collaborative information needed for efficiency, and (b) differing types of user collaboration behavior observed in the context of Web mining. To the best of our knowledge, this research is the first effort to develop a Web search system that supports collaboration by sharing complete search sessions containing post-retrieval analysis.

The rest of the article is outlined as follows: Section 2 reviews related research and commercial products. Section 3 describes the system architecture and main components of Collaborative Spider. In Section 4, a sample user session with the system is described in detail to illustrate how it performs Web mining and facilitates user collaboration and sharing. Sections 5 and 6 focus on a user study designed to test the effectiveness of Collaborative Spider and answer the research questions raised above. We conclude the paper in Section 7 by summarizing our research contributions and pointing out future research directions.

2. Related work

2.1. Web search engines

Many search engines are available on the Internet, each having its own characteristics and employing different algorithms to index, rank, and present Web documents. Examples of popular general-purpose search engines include AltaVista (http://www.altavista.com), Google (http://www.google.com), and Excite (http://www.excite.com). These search engines allow users to submit queries and present the returned Web pages in ranked order. In Yahoo! (http://www.yahoo.com), Web pages are manually grouped into categories to create a hierarchical directory of a subset of the Internet. There are also domain-specific search engines, such as LawCrawler (http://www.lawcrawler.com), which searches for legal information on the Web.

Because a single search engine can cover only a small portion of the Web [29], meta-search engines also have gained popularity. A meta-search engine is a system that forwards user queries to several search engines, aggregates the returned results, and presents the combined (usually re-ranked) results to the user. MetaCrawler (http://www.metacrawler.com), DogPile (http://www.dogpile.com) and 37.com (http://www.37.com) are such examples.

2.2. Web content mining and post-retrieval analysis

Web mining is the use of data mining techniques to automatically discover and extract information from Web documents and services [13]. Web mining research can be classified into three categories: Web content mining, Web structure mining, and Web usage mining [27]. Web content mining refers to the discovery of useful information from Web contents, including text, image, audio, video, etc. Research in Web content mining encompasses resource discovery from the Web [5,12], document categorization and clustering [25,48], and information extraction from Web pages [42]. Web structure mining studies the Web's hyperlink structure. It usually involves analysis of the in-links and out-links of a Web page, and it has been used for search engine result ranking [4,23]. Web usage mining focuses on analyzing search logs or other activity logs (in a way similar to data mining) to find interesting patterns. One of the main applications of Web usage mining is to learn user profiles [1,46].

The research reported in this paper falls into the category of Web content mining. When a typical Web search engine returns a ranked list of links to a set of Web pages relevant to a user query, the user then has to go through these Web pages manually to gain a comprehensive understanding of them and judge their relevance to the original query. This browsing process can be very time-consuming and requires substantial mental effort. Web content mining tech-niques can be applied to perform post-retrieval analysis of a retrieved document set that has been shown to provide the user with a quick impression of the set of retrieved Web pages and generally improve the searching experience [10,48]. Categorization and clustering techniques have been applied to classify search results into categories such that the user can browse and navigate through the set of retrieved pages more easily. NorthernLight (http://www.northernlight.com), a commercial search engine, categorizes retrieved Web pages into predefined search categories called *Custom Search Folders*. Another approach is to categorize Web pages on the fly without resorting to predefined categories. For example, the *Self-Organizing Map* (SOM) approach has been successfully applied to categorize retrieved Web pages into different regions on a 2-D topic map [9,24,25].

A major drawback of post-retrieval analysis is the computation time and resources needed. While a simple ranked list of search results usually can be returned to the user within a few seconds, post-retrieval analysis may take much longer, from several seconds up to a few minutes. Also, more computation, time, and memory are often required. These limitations may be severe, especially for Web-based search engines that have to handle thousands to millions of search queries per day [6].

2.3. Collaborative information retrieval and collaborative filtering

In order to alleviate the information overload problem that has resulted from the overwhelming volume of available Internet resources, collaborative information retrieval techniques have been proposed and studied in the context of *computer supported cooperative work* (CSCW) that allows multiple users to perform search collaboratively or to share their past search and analysis experiences. Sharing search results or better, sharing all the data about entire search sessions, constitute a basic requirement for a collaborative information retrieval system [22].

There are in general two approaches to collaborative information retrieval. The first is concerned with situations where several people utilize CSCW tools to support collaboration in the information retrieval process. Users collaboratively search for answers to the same query. Individual findings are then aggregated and unified by the users [2].

The second approach is what has been called *collaborative filtering* or *recommender systems*. Goldberg et al. [17] define collaborative filtering as collaboration in which people help one another perform filtering by recording their reactions to Web documents they read. Examples of collaborative filtering and recommender systems include Amazon.com, GroupLens [26], Fab [3], Ringo [38], and Do-I-Care [39]. When a user performs a search, these systems will recommend a set of documents or items that may be of interest based on that user's profile and other users' interests and past actions. For example, Amazon.com uses collaborative filtering to recommend books to potential customers, based on the preferences of other customers who have similar interests or purchasing histories. Annotations in free text or predefined formats are also incorporated in systems such as AntWorld [21], Annotate! [16], and CIRE [35] to facilitate collaborative retrieval among users.

Collaborative information retrieval systems can also help users with different backgrounds to share information more effectively. For example, the Worm Community System [7] helps users from different backgrounds to solve the vocabulary difference problem. Shank [37] also points out that for interdisciplinary participation on the Internet, not only information, but also world views are shared among users, creating a perfect environment for knowledge creation.

One of the major issues for collaborative information retrieval system is the users' willingness to share information. Orlikowski [33] observed that Lotus Notes was not well utilized because workers had little or no incentive to share information. The situation, however, becomes less problematic for Web search, which consists mostly of voluntary contributions [35]. Users are more willing to contribute in an exchange of pride and popularity. In addition, many systems try to minimize user effort by capturing user profiles and patterns automatically [1,39].

2.4. Software agents

2.4.1. Agents on the Web

Software agents have been widely used in Web applications. Web search agents known as *spiders* or *crawlers*, typically run by Web search engines, are used to fetch Web pages from Web servers. Another major type of Web agent, residing on user machines, helps a user search the Web and perform personalized information filtering and management. For example, Blue Squirrel's WebSeeker (http://www.bluesquirrel.com) and Copernic 2001 (http://www.copernic.com) connect with various search engines, monitor Web pages for any changes, and schedule automatic searches. Excalibur RetrievalWare and Internet Spider (http://www.excalib.com) collect, monitor, and index information from text documents on the Web as well as image files. Focused Crawler [5] locates Web pages relevant to a predefined set of topics based on example pages provided by the user. It also analyzes the link structures among the Web pages collected. The Itsy Bitsy Spider [8] searches the Web based on starting URLs and user-provided keywords using a genetic algorithm. Other AI algorithms, such as hybrid simulated annealing, also have been employed in Web search agents [47].

Because software agents can work autonomously and observe and learn from user actions, agent technology has also been applied in information filtering to reduce information overload. For example, the Maxims agent filters e-mail messages based on dynamically acquired user e-mail patterns [31]. NewsWeeder [28] helps the user filter Usenet Netnews by learning from examples. Many other collaborative filtering systems, notably GroupLens mentioned above, also utilize agent technology.

2.4.2. Multi-agent systems

A multi-agent system is one in which a number of agents cooperates and interact with each other in a complex and distributed environment. In a typical multi-agent system, each agent has an incomplete information or capabilities. The agents work together to achieve a global objective based on distributed data and control. In most cases, the interaction is asynchronous and decentralized. Jennings et al. [19] provide a detailed review of the field.

Multi-agent systems have been developed for a variety of application domains, including electronic commerce, air traffic control, workflow management, transportation systems, and Web applications, among others [40,41]. To enable effective inter-agent communication and coordination, agents that work together have to use an interoperable, platform-independent, and semantically unambiguous communication protocol. The two most widely used agent communication languages (ACL) are the Knowledge Query and Manipulation Language (KQML) and the FIPA ACL. KQML, developed as part of the ARPA Knowledge Sharing Effort, is a language and protocol for exchanging information and knowledge among software agents. In KQML, each expression is a *speech act* described by a *performative* [14,15]. The FIPA ACL was developed by the Foundation for Intelligent Physical Agents (FIPA). Similarly to KQML, the FIPA ACL is based on speech act theory and the two languages have similar syntax.

2.5. Problems in current approaches

In this section, we summarize the two major problems with current Web searching and mining approaches, which motivate our research and the Collaborative Spider system presented in this paper.

First, although real-time post-retrieval analysis has been proven effective for Web searching, very few systems perform it. For commercial Web search engines, this kind of analysis can be prohibitively expensive from a computational viewpoint. Recent research prototypes are starting to incorporate such analysis capability into client-side Web computing [6].

Second, for systems that do perform post-retrieval analysis, the analysis is based entirely on individual searches. In these systems, search sessions are not shared by the users. Search strategies, which may have taken a significant amount of time and effort to formulate and test, are lost when the related search session is complete. As a result, users are essentially on their own when they perform search tasks; they reinvent the wheel quite often, oblivious to potentially much improved Web searching and mining experience that could make collaboration among users possible.

To address these problems, we propose the Collaborative Spider system, which incorporates post-retrieval analysis and collaboration based on search session sharing. The main research issues explored include (a) the impact on users' search and analysis performance and the optimal volume of collaborative information needed for efficiency, and (b) different types of user collaboration behavior observed in the context of Web mining. The design and evaluation of the system will be discussed in the following sections.

3. Collaborative spider: system architecture and main components

The thesis of our research presented in this paper is that a collaborative Web information retrieval and mining environment that performs in-depth post-retrieval analysis leads to improved search effectiveness and efficiency. This idea is embodied in a multi-agent system that we have developed called the Collaborative Spider system. In this section, we present the architectural design and the main technical components of Collaborative Spider.

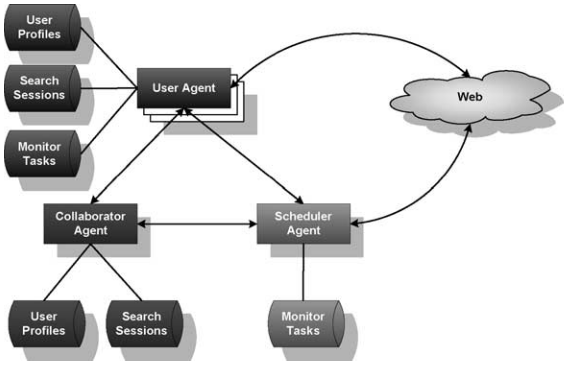


Fig. 1. Architecture of Collaborative Spider.

The system architecture is shown in Fig. 1. Collaborative Spider consists of three types of software agents, namely, *User Agent*, *Collaborator Agent*, and *Scheduler Agent*. In a typical system setup, each individual user will have his or her own personalized User Agent. Each user group (e.g., all participants of a research project or members of a new product design team) will share one Collaborator Agent and one Scheduler Agent. The User Agent is mainly responsible for retrieving pages from the Web, performing post-retrieval analysis, and interacting with the users. The Collaborator Agent facilitates the sharing of information among different User Agents. The Scheduling Agent keeps a list of monitoring tasks and is responsible for carrying out these tasks based on users' schedules.

Our architecture differs from traditional information retrieval systems or recommender systems in that collaboration is based on users' searches and analysis, not Web pages rated [3], news articles viewed [26], or items purchased (e.g., Amazon.com). The functionalities of each type of agent are discussed in detail in the following sections.

3.1. User agent

The User Agent is developed based on Competitive Intelligence (CI) Spider, a prototype system developed in our previous research [11]. CI Spider is a personalizable Web search tool designed to help the user search within a given Web site and perform post-retrieval analysis on the set of Web pages retrieved. Technically, the User Agent consists of four main components: *Internet Spiders*, *Arizona Noun Phraser*, *Self-Organizing Map*, and *Knowledge Dashboard*. The Internet Spiders perform breadth-first search or best-first search on Web sites specified by the user. The Web pages are then fetched to the user machine and passed to the *Arizona Noun Phraser* (AZNP) for further analysis. Developed at the University of Arizona, AZNP extracts and indexes all noun phrases from each document collected by the Internet Spiders based on part of speech tagging and linguistic rules [42]. The noun phrases are then presented to the user, ranked in descending order of occurrence frequency. If further analysis is desired, the data are aggregated and sent to the *Self-Organizing Map* (SOM) for automatic categorization and visualization. The SOM employs an artificial neural network algorithm to cluster the Web pages automatically into different regions on a 2-D topic map [9]. In SOM, more important concepts occupy larger regions and similar concepts are grouped together in a neighborhood [30]. This provides the user with a convenient and intuitive way to browse the most important topics in the result set.

The main interface component of the user agent is called a *Knowledge Dashboard*, which enables collaborative information search. It shows the information shared among users including detailed search results, analysis, and users' annotations. In addition to browsing past search sessions, the user can also launch new search and Web site monitoring tasks by choosing a combination of any information items and search criteria on the dashboard. More details are discussed in Section 4.

3.2. Scheduler agent

The Scheduler Agent runs continuously on the background listening for monitoring requests sent by the User Agent. It keeps a complete list of the monitoring tasks for every user and is responsible for carrying out each retrieval and analysis task according to a user-given schedule. The Scheduler Agent also performs load-balancing to avoid overloading the same Web server when there are a large number of scheduled tasks using simple heuristics (e.g., to prevent launching two scheduled search tasks simultaneously on the same Web site). Whenever a new search session is completed, the Scheduler Agent will store the session and forward it to the corresponding User Agent and Collaborator Agent.

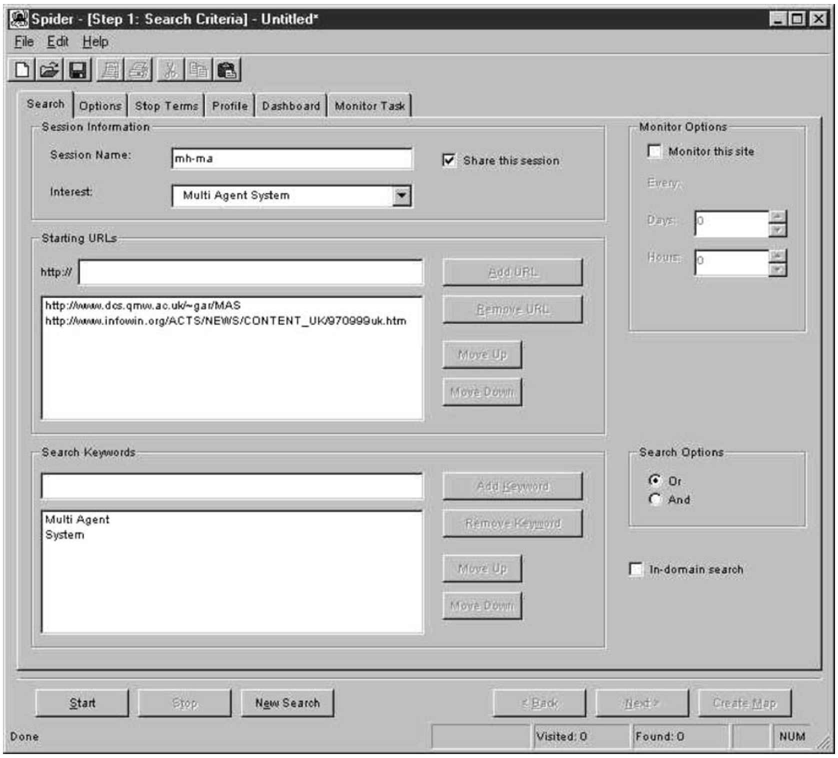
3.3. Collaborator agent

The Collaborator Agent is the central piece of Collaborative Spider. It functions as a mediator and regulates interactions between the User Agents and the Scheduler Agent. It also maintains a collective data repository, which stores all user search and monitoring sessions as well as user profiles. To ensure system robustness, each User Agent continues to store a subset of data associated with its user such that the system will be responsive even if the Collaborator Agent or the Scheduler Agent goes down. In addition to user search sessions and profiles, the Collaborator Agent keeps track of user annotations and comments that can be attached to any documents that the user has browsed or studied. These annotations are accessible to other users.

One of the key functionalities of the Collaborator Agent is to recommend Web documents to potentially interested users, based on profile matching. While different types of recommendation strategies may be used, a simple approach is used in the current implementation. Recommendations are made based on users' areas of interest. When a user performs a search, the search topic needs to be explicitly specified. This search session will then be shared with other users who have selected the corresponding area of interest in their profiles. Section 4 will provide more detailed examples to illustrate how the Collaborator Agent facilitates Web searching and mining experience sharing among multiple users.

3.4. Data repository design

Aimed at simplicity, the system stored all high-level data in plain-text file format. The system follows a simple relationship database design and all the tables follow a certain degree of database normalization. There are three main entities in the data repository, namely, *User Profiles*, *Search Sessions*, and *Monitor Tasks*. User Profiles hold information about the users of the system, including name, user id, email, and areas of interest, among other personal information. Search



Session Name: mh-ma

Interest: Multi Agent System

☑ Share this session

http://

http://www.dcs.qmw.ac.uk/~gar/MAS
http://www.infowin.org/ACTS/NEWS/CONTENT_UK/970999uk.htm

Multi Agent
System

□ Monitor this site

Days: 0

Hours: 0

● Or
○ And

□ In-domain search

Visited: 0 Found: 0

Fig. 2. Specifying starting URLs, search terms, and sharing options.

Sessions store the information about each search session performed. This information includes user id, session id, the area each session belongs to, date, and time of the search, whether the search is shared, starting URLs and search terms used and other search options, and the comments and annotations by other users. Monitor Tasks store all the Web site monitoring tasks specified by the users. The data stored include user id, session id, date, and the frequency for revisit. One should note that the lists of information discussed are not exhaustive; they can be easily expanded to accommodate future system enhancement (e.g., to support anonymity).

3.5. Agent communication language

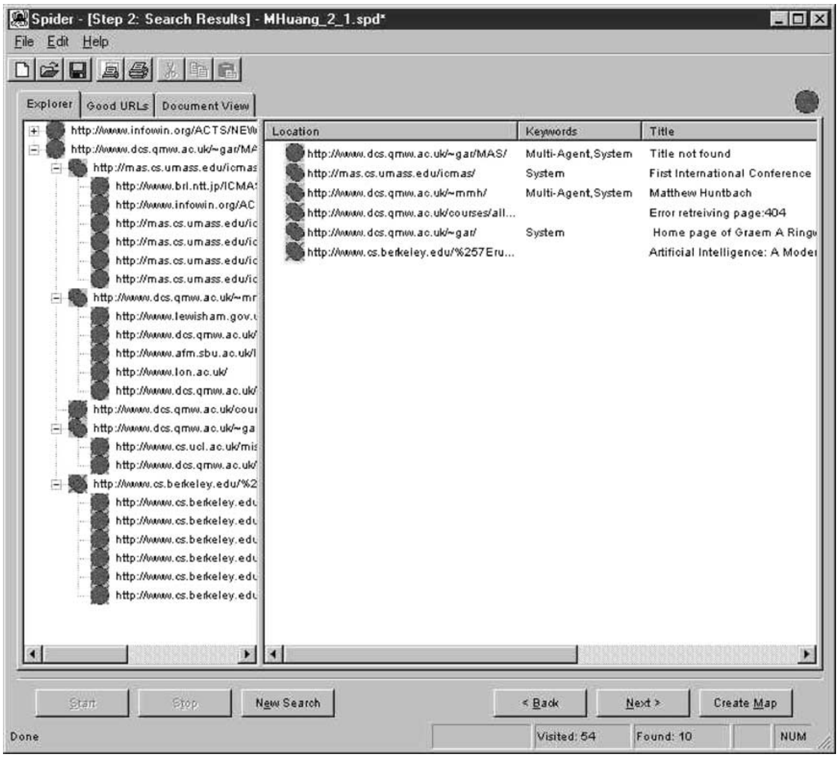
The Knowledge Query and Manipulation Language (KQML) is used as the communication language by the agents in the Collaborative Spider system. We use JATLite (Java Agent Template, Lite) as the KQML implementation platform. JATLite, developed at the Center for Design Research at Stanford University, is a Java implementation of KQML [20,34]. Agents send and receive KQML messages in our system through a *message router* using TCP/IP protocol. The use of agent templates facilitates agent development by aggregating common functions and services across all agent classes into a few abstract classes.

4. Sample user sessions using Collaborative Spider

We provide detailed examples in this section to illustrate how a user interacts with Collaborative Spider.

4.1. User registration

First-time users are required to register with Collaborative Spider through a User Agent. The user must select at least one area of interest from the *Areas of Interest* panel. Examples of areas of interest include



```
+ http://www.infowin.org/ACTS/NEW
- http://www.dcs.qmw.ac.uk/~gar/MA
    - http://mas.cs.umass.edu/icmas
        http://www.brl.ntt.jp/ICMA:
        http://www.infowin.org/AC
        http://mas.cs.umass.edu/ic
        http://mas.cs.umass.edu/ic
        http://mas.cs.umass.edu/ic
        http://mas.cs.umass.edu/ic
    - http://www.dcs.qmw.ac.uk/~mr
        http://www.lewisham.gov.u
        http://www.dcs.qmw.ac.uk/
        http://www.afm.sbu.ac.uk/l
        http://www.lon.ac.uk/
        http://www.dcs.qmw.ac.uk/
      http://www.dcs.qmw.ac.uk/cou
    - http://www.dcs.qmw.ac.uk/~ga
        http://www.cs.ucl.ac.uk/mis
        http://www.dcs.qmw.ac.uk/
    - http://www.cs.berkeley.edu/%2
        http://www.cs.berkeley.edu
        http://www.cs.berkeley.edu
        http://www.cs.berkeley.edu
        http://www.cs.berkeley.edu
        http://www.cs.berkeley.edu
        http://www.cs.berkeley.edu
```

Location	Keywords	Title
http://www.dcs.qmw.ac.uk/~gar/MAS/	Multi-Agent,System	Title not found
http://mas.cs.umass.edu/icmas/	System	First International Conference
http://www.dcs.qmw.ac.uk/~mmh/	Multi-Agent,System	Matthew Huntbach
http://www.dcs.qmw.ac.uk/courses/all...		Error retreiving page:404
http://www.dcs.qmw.ac.uk/~gar/	System	Home page of Graem A Ringw
http://www.cs.berkeley.edu/%257Eru...		Artificial Intelligence: A Moder

Fig. 3. Search results.

Information Visualization, *Expert System*, and *Data Mining*. After receiving the user input, the User Agent updates its local data source as well as the collective data repository managed by the Collaborator Agent.

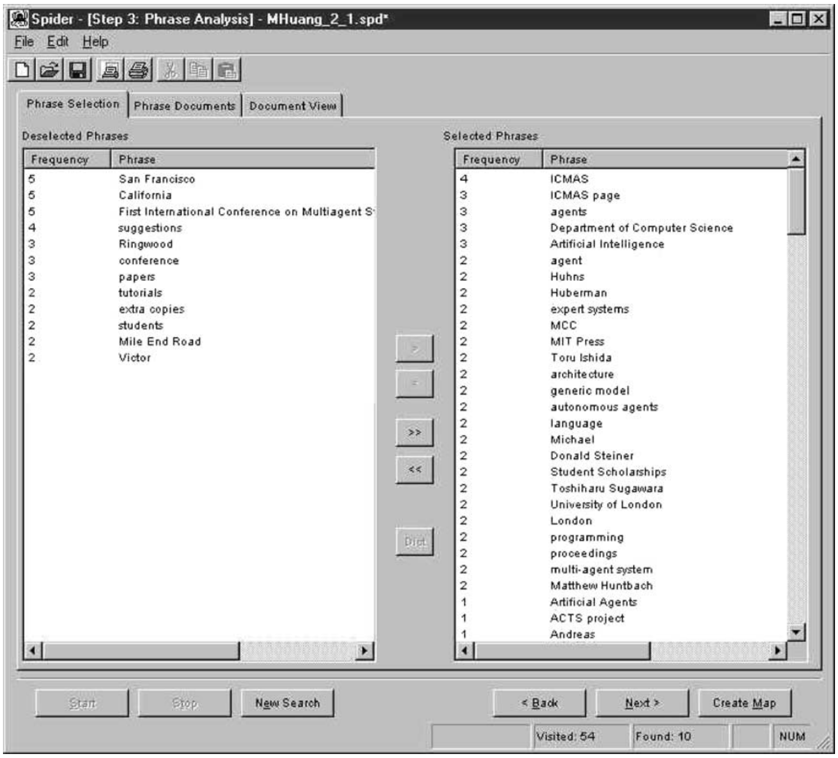
4.2. Web search and analysis

After user registration and profile specification, the user is ready to perform searches. As shown in Fig. 2, the user can specify a *session name*, select the proper *areas of interest*, and have the option of whether or not to share this session with others who have the same interests. The next step is to add the *starting URLs* and *query terms*, such that the User Agent can perform a search based on the given information. The starting URLs specify the Web sites that the user wants to analyze.

Because the system uses a breadth-first search algorithm, the Web sites specified will be searched exhaustively or until the required number of Web pages has been collected. Because starting URLs specify the Web sites from which information should be retrieved, different staring URLs can result in completely different search results. In the given example, the User Agent performed a search on *Multi-Agent* and *System* on the two Web sites.

Whenever a page is collected during the search, the link to that page is displayed dynamically on one of the result screens (see Fig. 3). The left frame in Fig. 3 shows a hierarchical structure of the Web pages visited. When the user clicks on a link on the left, the link and all the links contained in that Web page will be displayed in the right frame of the result window. The user can then click on any link displayed and read the full-text content without having to wait for the whole search to be completed. To facilitate user browsing, all pages that contain the search keyword(s) are marked by a globe icon without the red cross, as illustrated in Fig. 3.

In addition to performing the search, the User Agent also provides post-retrieval analysis on the fly for the user. If the user clicks on the *Next* button in the



Deselected Phrases

Frequency	Phrase
5	San Francisco
5	California
5	First International Conference on Multiagent S
4	suggestions
3	Ringwood
3	conference
3	papers
2	tutorials
2	extra copies
2	students
2	Mile End Road
2	Victor

Selected Phrases

Frequency	Phrase
4	ICMAS
3	ICMAS page
3	agents
3	Department of Computer Science
3	Artificial Intelligence
2	agent
2	Huhns
2	Huberman
2	expert systems
2	MCC
2	MIT Press
2	Toru Ishida
2	architecture
2	generic model
2	autonomous agents
2	language
2	Michael
2	Donald Steiner
2	Student Scholarships
2	Toshiharu Sugawara
2	University of London
2	London
2	programming
2	proceedings
2	multi-agent system
2	Matthew Huntbach
1	Artificial Agents
1	ACTS project
1	Andreas

Visited: 54
Found: 10

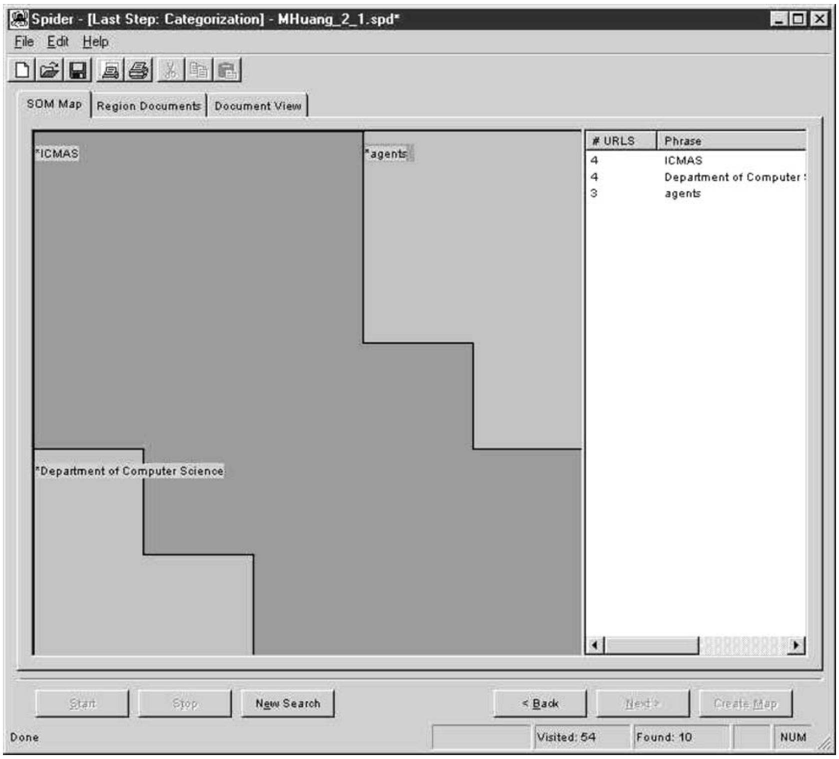
Fig. 4. Noun phraser showing the terms that appear most frequently in the Web pages retrieved.

Search Results screen, the Arizona Noun Phraser will provide a list of the noun phrases that appear most frequently in the set of documents retrieved (see Fig. 4). The user can view the document occurrence frequency of each phrase and is provided links to the documents containing that phrase. If the user wants to perform further analysis, he or she can select the phrases of interest to generate a 2-D topic map produced by the Self-Organizing Map (SOM) algorithm. The topic map provides the user with an overview of the set of Web pages collected, as shown in Fig. 5. In our example, the Web pages retrieved were clustered into three categories, namely, *ICMAS*, *Agents*, and *Department of Computer Science*.

4.3. Accessing other users' search sessions

The *Knowledge Dashboard* panel can be used by a user to access other users' search sessions (see Fig. 6). Search sessions in the user's areas of interest will be displayed to the user. In this case, users *Michael Huang*, *Michael Chau,* and *Daniel Zeng* all were interested in the *Multi-Agent System* area. Should the user decide to use some of the *URLs* or *search terms* from past search sessions, he or she can click on the items preferred (a red check mark will appear) and then press the *Add to Query* button. All the selected items will be added to the search panel along with those already input. The user can also start a brand new search with the help of the collaborative information obtained from the dashboard.

In addition to using other users' URLs or Keywords, the user can also choose to load a complete session performed by another user from the dashboard. The user can identify the session of interest based on the session names entered by other users. When a descriptive name is not available (as in the example), the user can choose the session based on other users' reputation or the keywords and URLs used by them. A search session can be retrieved by highlighting the session of interest and clicking the *Load* button. As shown in Fig. 6, the user also has the option to add



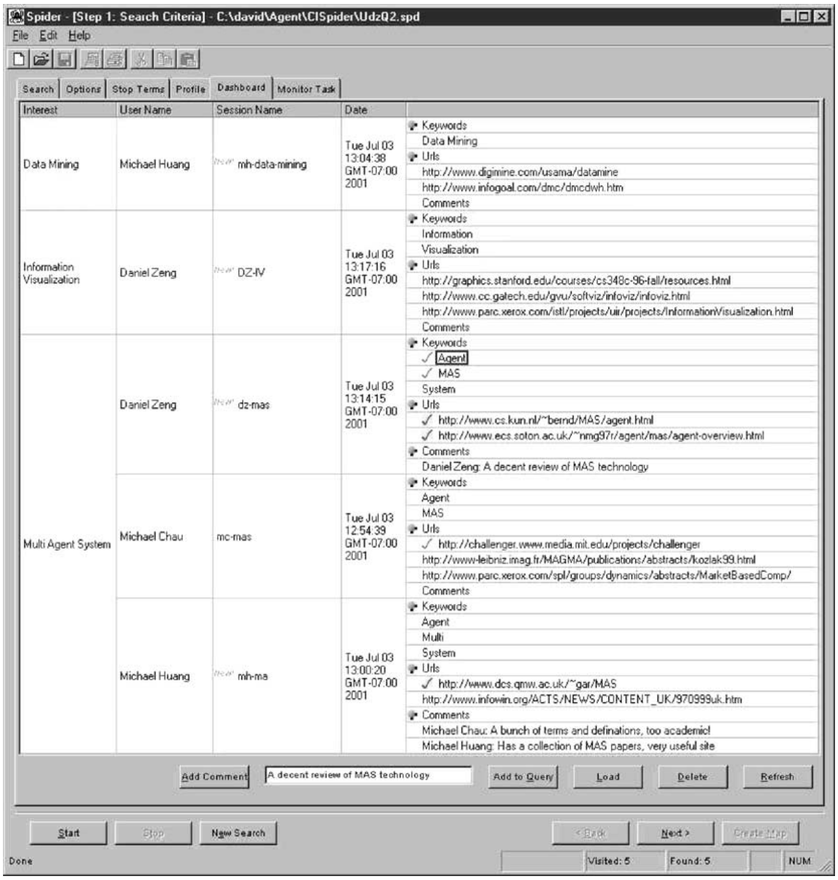
# URLS	Phrase
4	ICMAS
4	Department of Computer :
3	agents

Done

Visited: 54

Found: 10

Fig. 5. Self-Organizing Map (SOM), a topic map that categorizes the retrieved Web pages into different regions.



Interest	User Name	Session Name	Date	
Data Mining	Michael Huang	mh-data-mining	Tue Jul 03 13:04:38 GMT-07:00 2001	Keywords Data Mining Urls http://www.digimine.com/usama/datamine http://www.infogoal.com/dmc/dmcdwh.htm Comments
Information Visualization	Daniel Zeng	DZ-IV	Tue Jul 03 13:17:16 GMT-07:00 2001	Keywords Information Visualization Urls http://graphics.stanford.edu/courses/cs348c-96-fall/resources.html http://www.cc.gatech.edu/gvu/softviz/infoviz/infoviz.html http://www.parc.xerox.com/istl/projects/uir/projects/InformationVisualization.html Comments
Multi Agent System	Daniel Zeng	dz-mas	Tue Jul 03 13:14:15 GMT-07:00 2001	Keywords ✓ Agent ✓ MAS System Urls ✓ http://www.cs.kun.nl/~bernd/MAS/agent.html ✓ http://www.ecs.soton.ac.uk/~nmg97r/agent/mas/agent-overview.html Comments Daniel Zeng: A decent review of MAS technology
	Michael Chau	mc-mas	Tue Jul 03 12:54:39 GMT-07:00 2001	Keywords Agent MAS Urls ✓ http://challenger.www.media.mit.edu/projects/challenger http://www-leibniz.imag.fr/MAGMA/publications/abstracts/kozlak99.html http://www.parc.xerox.com/spl/groups/dynamics/abstracts/MarketBasedComp/ Comments
	Michael Huang	mh-ma	Tue Jul 03 13:00:20 GMT-07:00 2001	Keywords Agent Multi System Urls ✓ http://www.dcs.qmw.ac.uk/~gar/MAS http://www.infowin.org/ACTS/NEWS/CONTENT_UK/970999uk.htm Comments Michael Chau: A bunch of terms and definations, too academic! Michael Huang: Has a collection of MAS papers, very useful site

Add Comment: A decent review of MAS technology

Fig. 6. Knowledge dashboard.

comments to his or her own session or any other sessions shown on the dashboard.

In the current system, the identity of a user is automatically captured by the system and shown to other users. In future implementation, we expect to allow a user to choose whether his/her name should be hidden from others, as some users may prefer to be anonymous.

4.4. Saving and sharing search sessions

After finishing a search session, the user may save it in a binary file and share it with other users. Saving a search session will trigger the User Agent to send a message to the Collaborator Agent along with the real content of the saved session. The Collaborator Agent will then forward the metadata to all other users interested in the same area. Any user connected (logged on) to Collaborative Spider can immediately view this new session on the knowledge dashboard panel.

4.5. Monitoring sessions

The user can request the system to make regular visits to certain Web sites. Such requests are forwarded to the Scheduler Agent, which will constantly check the monitor tasks list and look for outstanding tasks. If an outstanding task is found, the Scheduler Agent will activate a number of Internet Spiders to conduct the search. After the search is completed, the Scheduler Agent will send a message to the Collaborator Agent to inform it of the new search session. The Collaborator Agent in turn will forward the metadata of the search session to all interested users.

5. Evaluation methodology

The evaluation study was designed to answer the following questions: (1) Does a collaborative Web information retrieval and mining environment that performs in-depth post-retrieval analysis lead to improved search effectiveness and efficiency? (2) How does the size of shared repository affect a user's Web search and mining effectiveness and efficiency?

5.1. Experimental design

In order to study the effect of collaboration on Web search and analysis, a user study was conducted. A key research issue was to explore the impact of the size of the shared repository on a user's Web search and mining effectiveness and efficiency. We hypothesized that a user's performance would gradually improve for the first few sessions, when more search sessions are made available. Beyond a certain number of search sessions, we hypothesized that the marginal benefit of additional search sessions would approach zero. To test our hypotheses, the User Agent and the Collaborator Agent, but not the Scheduler Agent, were evaluated in the user study.

The general design of the experiment was based on the theme-based evaluation framework previously developed in the evaluation of the CI Spider and the Meta Spider systems [6,10]. We gave each subject several relatively open-ended information search tasks and asked him or her to summarize search results in the form of a number of themes, rather than to find specific information. In our experiments, a theme was defined as "a short phrase that summarizes a specific aspect of the search results." Within this theme-based framework, we designed protocols to permit evaluation of the extent to which post-retrieval analysis and collaboration facilitate users' identification of major themes related to a certain topic.

Fifty undergraduate students, most of them majoring in management information systems, were recruited for the experiment. Six of the 50 topics used in the Sixth Text Retrieval Conference (TREC-6) ad hoc task were selected and modified for use in the context of Web searching [45]. The TREC series was sponsored by the National Institute of Standards and Technology (NIST) and the Defense Advanced Research Projects Agency (DARPA) to encourage research in information retrieval from large text collections. Each subject was given three search topics and asked to perform search and analysis to identify the major themes related to each of them.

The six topics we used in our experiments were:

1. Hubble telescope achievement
2. Implant dentistry
3. Radio waves and brain cancer
4. Undersea fiber optic cable
5. New fuel sources
6. Health and computer terminals

For each search task, a subject could choose one or more search strategies from the following list to perform an analysis using the Collaborative Spider system:

1. The subject could start a completely new search by entering new starting URLs and search terms, and analyze the search results. The subject could obtain the starting URLs using any search engine of his or her choice.
2. The subject could start a new search by using a combination of URLs and search terms from the knowledge dashboard, plus the subject's own starting URLs (obtained from any preferred search engine) and search terms.
3. The subject could start a new search by using URLs and search terms from the knowledge dashboard, if available. These URLs and search terms will be those previously used by other subjects.
4. The subject could browse the search sessions performed by other subjects (if available) and come up with the findings without actually doing a Web search.

The 50 subjects were equally divided into five groups. Each group had a different number of previous sessions available. Each subject in Group 0 had no access to search sessions performed by other subjects and was required to perform the search individually. Subjects in Group 1, for each search topic, had access to one previous search session for the same topic by another subject. These were search sessions performed by subjects in Group 0. Similarly, subjects in Group 2 had access to two search sessions performed and saved by subjects in Group 0 and Group 1, and subjects in

Group 3 had access to three sessions, etc. Based on a rotation scheme, half of the subjects within each group were given topics 1, 2, and 3 and half of the subjects were given topics 4, 5, and 6. This allowed us to control the amount of collaborative information available and measure its effect on the efficiency and effectiveness on Web search and analysis. An alternative design would be to assign each subject to different groups for different search tasks. Although such design might decrease the effect of subject bias on the data, it was not chosen because it would be much more difficult to control the experiment environment.

5.2. Performance measures

We recruited two graduate students majoring in library science as expert judges in our experiment. The expert judges individually performed extensive searches on the six search topics and summarized their findings into topic themes. Their results were then condensed and combined to form the basis for evaluation. Precision and recall rates for the number of themes were used to measure the effectiveness of each search performed and were calculated as follows:

$$\text{precision} = \frac{\text{number of correct themes identified by the subject}}{\text{total number of themes identified by the subject}}$$

$$\text{recall} = \frac{\text{number of correct themes identified by the subject}}{\text{total number of themes identified by expert judges}}$$

A well-accepted single measure that tries to balance recall and precision called F-measure was also used in our evaluation and calculated as follows [43]:

$$F\text{-measure} = \frac{\text{recall*precision}}{(\text{recall} + \text{precision})/2}$$

The F-measure value was calculated for each search session and the average value was used for each group.

We recorded the amount of time each subject spent for each search topic. During the experiment, we encouraged our subjects to tell us about the search method used and their comments were recorded. The experimenter (one of the co-authors) also closely observed each test subject during the experiment and filled out an observation log to record the user actions. At the end of the experiment, each subject filled out a questionnaire to offer further comments on the search system and the search strategies used.

6. Experimental results and discussions

6.1. Quantitative results

The average times spent on the set of the search tasks by the subjects in different groups are summarized in Table 1. No significant difference in total search time was observed among the groups. The results show that the subjects were not able to reduce the total time by browsing or using other users' sessions. By looking at the amount and percentage of time spent on each particular type of search and mining activity, we found that subjects across the groups spent comparable amounts of time on these activities, except for Group 0, to whom other users' sessions were not available.

The results on average precision, recall, and F-measures are summarized in Table 2. Each group represents a sample size of 30 subject–task combinations. A corresponding chart is shown in Fig. 7. We also used Group 0 (where subjects had no access to any collaborative information) as a reference group for comparison with other groups. A series of t-tests were conducted, the results are shown in Table 3.

Comparing the performances of Group 0 and Group 4, we found that the average precision, recall, and F-

Table 1
Average time spent on each search task

Group	0	1	2	3	4
Total search time (min)	16.1 (100%)	15.3 (100%)	15.8 (100%)	16.6 (100%)	16.3 (100%)
Time spent for each subtask (min)					
Getting URLs from search engines	6.8 (42.2%)	4.3 (28.2%)	5.4 (34.3%)	5.6 (34.0%)	5.5 (33.7%)
Browsing metadata on the dashboard	0.0 (0.0%)	1.2 (7.7%)	1.2 (7.4%)	1.3 (7.6%)	1.8 (10.8%)
Browsing other users' session(s)	0.0 (0.0%)	1.3 (8.3%)	1.0 (6.1%)	0.5 (3.6%)	1.7 (10.6%)
Performing own search and analysis	9.3 (57.8%)	8.5 (55.8%)	8.3 (52.2%)	9.1 (55.2%)	7.3 (44.9%)

Table 2
Analysis of effectiveness on different groups

Group	0	1	2	3	4
Sample size	30	30	30	30	30
Average precision	0.474	0.462	0.503	0.557	0.569
Average recall	0.243	0.217	0.211	0.280	0.312
Average *F*-measure	0.299	0.275	0.282	0.355	0.387

Table 3
p-values of *t*-tests on groups' performances

	Group 0 vs. Group 1	Group 0 vs. Group 2	Group 0 vs. Group 3	Group 0 vs. Group 4
Precision	0.859	0.752	0.268	0.150
Recall	0.527	0.487	0.362	0.078 *
F-measure	0.569	0.750	0.246	0.070 *

* Significant at the 10% level.

measure climbed to 0.569, 0.312, 0.387 in Group 4, in which subjects had the most collaborative information, from 0.474, 0.243, 0.299 in Group 0, in which subjects had no collaboration at all (see Table 2). From the *t*-test results in Table 3, we see that the differences between the recall rates and the *F*-measures of the two groups were statistically significant at the 10% level, indicating that the collaboration provided by the system was able to improve users' search performance.

In order to study the effect of size of the shared repository on search performance, we analyzed the performance of each group in detail. In Fig. 7, the three leftmost data points indicate the performance of subjects who had no access to any other subjects' search sessions (Group 0). Search and analysis performance started to decline for subjects having access to one other user's search session (Group 1) and for those having access to two other users' search sessions (Group 2). As shown in Table 3, the precision, recall, and *F*-measure of these two groups were not statistically different from those of Group 0. The results demonstrate that subjects were unable to achieve improvement from having access to one or two search sessions by other users. We observed that when subjects had only a small amount of collaborative information, the normally expected improvement in performance did not occur. We believe this was because the amount of information available was so small (only one or two search sessions with a few URLs and search terms) that the subjects were not able to benefit appreciably from the collaboration. The effort and attention that the subjects expended on collaboration counteracted any performance gain from a limited amount of collaborative information. Sometimes, subjects might even have been distracted or confused by such information. As a result, in Groups 1 and 2, the performance level actually dropped slightly below that of Group 0, in which subjects had no collaborative information at all.

Proceeding from Group 2 to Group 3, it can be seen that there were notable improvements in all the three performance measures, and the search perform-

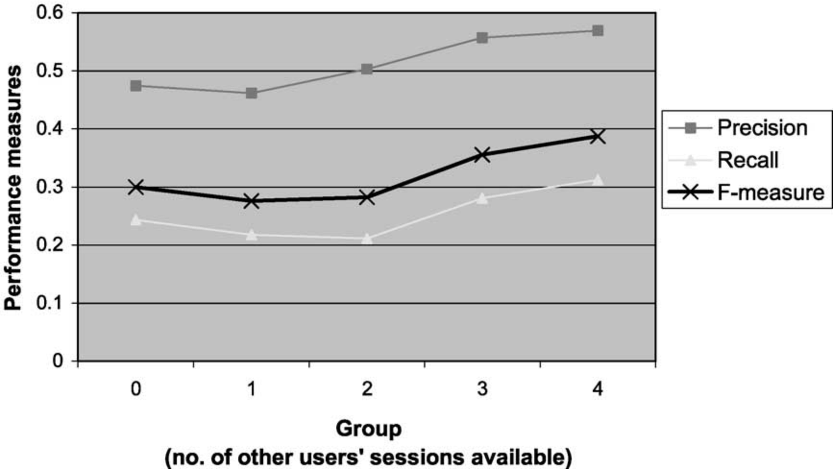


Fig. 7. Performance measures vs. the number of other users' sessions available.

ance further increased and reached a plateau for Group 4. This may have been because subjects started to benefit from other users' searches when they had access to three or more sessions. Previous sessions could be compared and benefits derived from the best of them, including help in deciding which sessions to explore, based on the annotations made by other subjects. We suggest that at this point the benefit of accessing other users' sessions started to outweigh the overhead cost of time and effort spent reading other users' sessions. Nevertheless, we suspect that after a certain threshold, the marginal benefit of having more previous search sessions available will diminish.

6.2. Collaboration behavior

In this section, we present a taxonomy of various types of collaboration behavior based on the search strategies observed during the experiment. We also discuss some qualitative data and technical insights obtained from the post-search questionnaires.

Because the subjects in Group 0 did not have access to other users' search sessions, we focused our analysis on the 120 cases (10 subjects each from Groups 1 through 4, with each subject performing three search queries) in which subjects could browse or load other users' sessions. We observed that most subjects did not attempt to rely solely on the search sessions available; they tried to combine other users' URLs and keywords with their own and launch new searches. In fact, of the 120 cases, in only 4 (1.7%) did the subject simply load other users' search sessions and come up with findings without actually performing a search. There were 34 cases (28.3%) in which subjects combined findings from their own searches with those from other users' search. In 57 cases (47.5%), subjects combined other users' starting URLs and search terms and launched new searches, using only the other users' starting points but not their findings. This method was found to be the one most commonly used in our experiment. It is interesting to note that although the subjects realized the value of sharing and collaboration and gained from obtaining some useful information from other users, they preferred to perform their own search sessions and draw their own conclusions. In the remaining 25 cases (20.8%), subjects performed their own searches without using any of the information available from other search sessions. To summarize, we categorize the types of collaboration behavior observed in our experiment as follows:

(A) Use own starting points (URLs and search terms) and perform new analysis.
(B) Combine other users' starting points with own starting points and perform new analysis.
(C) Use starting points of other users and perform new analysis.
(D) Use other users' starting points and their analysis.

We have analyzed the performance for each type of collaboration and summarized the results in Table 4.

From Table 4, we observed that the performance measures were comparable for Types A, B and C (the performance of Type D was not considered, given the small sample size). However, Type B, i.e., using a combination of their own starting points (URLs and search terms) and those of other users, was the most preferred search strategy. This strategy is in fact similar to the follow-up searches triggered by search results as discussed by O'Day and Jeffries [32]. Such follow-up searches often intend to probe more deeply in the same topic area. We believe that the use of other users' starting points reassured the subject that he or she was not off-track in performing the search. At the same time, subjects also liked to add their own starting points rather than relying entirely on other users' findings. This strategy appears to have provided a productive balance between having individual control over the search task and gaining help from other users.

Table 4
Performance analysis of different types of collaboration behavior

Type	A	B	C	D
Total number of cases	25 (20.8%)	57 (47.5%)	34 (28.3%)	4 (1.7%)
Number of subjects who considered this behavior as most efficient	6 (15.0%)	26 (65.0%)	7 (17.5%)	1 (2.5%)
Precision	0.522	0.486	0.516	0.321
Recall	0.268	0.228	0.252	0.163
F-measure	0.334	0.294	0.321	0.215

7. Conclusion and future directions

In this paper, we present our work on the design and evaluation of a multi-agent collaborative Web mining system. The main contribution of this research was the development and evaluation of the first Web search system that supports collaboration by sharing complete search sessions based on post-retrieval analysis. We demonstrated the feasibility of using a multi-agent architecture to build a collaborative Web searching and mining environment with session sharing. An initial evaluation study was designed and conducted to investigate the correlation between search performance and the amount of collaborative information available. Our study showed that subjects' performances were slightly degraded from those of individual search situations when they had access to one or two other users' search sessions but improved significantly when subjects had access to three or more search sessions of other users. We believe that in our system having three or more sessions available is the point at which the gain from collaboration outweighed the overhead of browsing and comprehending other users' sessions. In summary, we seem to have found our system's *point of insufficiency*, less than which the amount of collaborative information will be considered insufficient to provide performance gain in excess of the overhead expended to acquire it. However, we have not found the *point of sufficiency*, beyond which performance gain from collaboration will become stable even if more collaborative information is available. It will be interesting to find out whether performance continuously improves as the number of previous sessions is increased beyond 4, or whether the performance will reach an optimal point.

We also catalogued the types of collaboration behavior observed in our user study. We noticed that a large proportion of the subjects enjoyed using some starting Web sources obtained from other users' past searches as guidance but retained control over their own search process and drew their own conclusions. Nevertheless, we found that different behaviors did not have a considerable impact on subjects' performance.

The nature of the tasks the subjects performed may have had a substantial impact on their collaborative behavior and effectiveness. We are currently planning a user study in which the subjects will be asked to search for specific answers to well-defined questions, as opposed to the open-ended soft search queries discussed in this paper. Because subjects can probably gain from easy access to concrete answers obtained by other users, it will be interesting to examine whether users will demonstrate collaboration behavior different from what we observed in this experiment.

Another future research plan is to perform data mining on user search activities such that user profiles can be learned automatically. Currently, users have to specify their areas of interest explicitly in order to access shared search sessions. We are currently planning to use data mining algorithms to enhance the Collaborative Spider system by including more sophisticated content-based or collaborative-based information recommendation functionalities.

In conclusion, we believe that the experimental results are interesting and useful for related research, and that the research issues identified should be further studied in other collaborative environments.

Acknowledgements

This project was partly supported by the following research grants:

- NSF Digital Library Initiative-2, "High-performance Digital Library Systems: From Information Retrieval to Knowledge Management," IIS-9817473, April 1999–March 2002.
- NSF/CISE/CSS, "An Intelligent CSCW Workbench: Analysis, Visualization, and Agents," IIS-9800696, June 1998–June 2001.

We would like to thank the expert judges and the subjects at the University of Arizona who participated in the user study. We would also like to thank Ming Yin of the Artificial Intelligence Lab for assisting in system implementation.

References

[1] R. Armstrong, D. Freitag, T. Joachims, T. Mitchell, Webwatcher: a learning apprentice for the World Wide Web, Proceedings of the AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments, Mar 1995.

[2] R. Baeza-Yates, J.A. Pino, A first step to formally evaluate collaborative work, Proceedings of the International ACM SIGGROUP Conference on Supporting Group Work: The Integration Challenge, Phoenix, AZ, Nov 1997.

[3] M. Balabanovic, Y. Shoham, Fab: content-based, collaborative recommendation, Communications of the ACM 40 (3) (1997) 66–72.

[4] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, Proceedings of the 7th International World Wide Web Conference (WWW7), Brisbane, Australia, Apr 1998.

[5] S. Chakrabarti, M. van der Berg, B. Dom, Focused crawling: a new approach to topic-specific web resource discovery, Proceedings of the 8th International World Wide Web Conference (WWW8), Toronto, Canada, May 1999.

[6] M. Chau, D. Zeng, H. Chen, Personalized spiders for web search and analysis, Proceedings of the First ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'01), Roanoke, VA, Jun 2001.

[7] H. Chen, Collaborative systems: solving the vocabulary problem, IEEE Computer (May 1994) 58–66.

[8] H. Chen, Y. Chung, M. Ramsey, C.C. Yang, An intelligent personal spider (agent) for dynamic internet/intranet searching, Decision Support Systems 23 (1) (1998) 41–58.

[9] H. Chen, A. Houston, R. Sewell, B. Schatz, Internet browsing and searching: user evaluations of category map and concept space techniques, Journal of the American Society for Information Science 49 (7) (1998) 582–603, Special Issue on AI Techniques for Emerging Information Systems Applications.

[10] H. Chen, H. Fan, M. Chau, D. Zeng, MetaSpider: meta-searching and categorization on the web, Journal of the American Society for Information Science and Technology 52 (13) (2001) 1134–1147.

[11] H. Chen, M. Chau, D. Zeng, CI spider: a tool for competitive intelligence on the web, Decision Support Systems 34 (1) (2002) 1–17.

[12] J. Cho, H. Garcia-Molina, L. Page, Efficient crawling through URL ordering, Proceedings of the 7th International World Wide Web Conference (WWW7), Brisbane, Australia, Apr 1998.

[13] O. Etzioni, The World Wide Web: quagmire or gold mine, Communications of the ACM 39 (11) (1996) 65–68.

[14] T. Finin, R. Fritzson, D. McKay, A language and protocol to support intelligent agent interoperability, Proceedings of the CE and CALS Washington 92 Conference, Jun 1992.

[15] T. Finin, R. Fritzson, D. McKay, R. McEntire, KQML as an agent communication language, Proceedings of the Third International Conference on Information and Knowledge Management (CIKM'94), Nov 1994.

[16] M. Ginsburg, Annotate! a tool for collaborative information retrieval, Proceedings of the 7th IEEE International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE'98), IEEE CS, 75–80, Los Alamitos, CA, 1998.

[17] D. Goldberg, D. Nichols, B. Oki, D. Terry, Using collaborative filtering to weave an information tapestry, Communications of the ACM 35 (12) (1992) 61–69.

[18] M.A. Hearst, J.O. Pedersen, Reexamining the cluster hypothesis: scatter/gather on retrieval results, Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'96), 1996, pp. 76–84.

[19] N. Jennings, K. Sycara, M. Wooldridge, A roadmap of agent research and development, Autonomous Agents and Multi-Agent Systems 1 (1998) 7–38.

[20] H. Jeon, C. Petrie, M. Cutkosky, JATLite: a Java agent infrastructure with message routing, IEEE Internet Computing 4 (2) (2000) 87–96.

[21] P.B. Kantor, E. Boros, B. Melamed, V. Meñkov, B. Shapira, D.J. Neu, Capturing human intelligence in the net, Communications of the ACM 43 (8) (2000) 112–115.

[22] M. Karamuftuoglu, Collaborative information retrieval: toward a social informatics view of IR interaction, Journal of the American Society for Information Science 49 (12) (1998) 1070–1080.

[23] J. Kleinberg, Authoritative sources in a hyperlinked environment, Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms, Baltimore, MD, USA, Jan 1999, pp. 668–677.

[24] T. Kohonen, Self-Organizing Maps, Springer, Berlin (1995).

[25] T. Kohonen, S. Kaski, K. Lagus, J. Salojärvi, V. Paatero, A. Saarela, Self-organization of a massive document collection, IEEE Transactions on Neural Networks 11 (3) (2000) 574–585, Special Issue on Neural Networks for Data Mining and Knowledge Discovery.

[26] J.A. Konstan, B. Miller, D. Maltz, J. Herlocker, L. Gordon, J. Riedl, GroupLens: applying collaborative filtering to Usenet news, Communications of the ACM 40 (3) (1997) 77–87.

[27] R. Kosala, H. Blockeel, Web mining research: a survey, ACM SIGKDD Explorations 2 (1) (2000) 1–15.

[28] K. Lang, NewsWeeder: learning to filter Netnews, Proceedings of the 12th International Conference on Machine Learning, San Francisco, CA, 1995.

[29] S. Lawrence, C.L. Giles, Accessibility of information on the Web, Nature 400 (1999) 107–109.

[30] C. Lin, H. Chen, J. Nunamaker, Verifying the proximity and size hypothesis for self-organizing maps, Journal of Management Information System 16 (3) (2000) 61–73.

[31] P. Maes, Agents that reduce work and information overload, Communications of the ACM 37 (7) (1994) 31–40.

[32] V.L. O'Day, R. Jeffries, Information artisans: patterns of result sharing by information searchers, Proceedings of the ACM Conference on Organizational Computing Systems (COOCS'93), 98–107, Milpitas, CA, Nov 1993.

[33] W. Orlikowski, Learning from notes: organizational issues in groupware implementation, Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'92), 1992, pp. 362–369.

[34] C. Petrie, Agent-based engineering, the web, and intelligence, IEEE Expert 11 (6) (1996) 24–29.

[35] N. Romano, D. Roussinov, J.F. Nunamaker, H. Chen, Collaborative information retrieval environment: integration of information retrieval with group support systems, Proceedings of the 32nd Hawaii International Conference on System Sciences (HICSS-32), 1999.

[36] E. Selberg, O. Etzioni, The MetaCrawler architecture for resource aggregation on the web, IEEE Expert 12 (1) (1997) 8–14.

[37] G. Shank, Abductive multiloguing, the semiotic dynamics of

navigating the net, The Arachnet Electronic Journal on Virtual Culture 1 (1) Mar 1993.

[38] U. Shardanand, P. Maes, Social information filtering: algorithms for automating "word of mouth", Proceedings of the ACM Conference on Human Factors and Computing Systems, Denver, CO, May 1995.

[39] B. Starr, M. Ackerman, M. Pazzani, Do-I-Care: a collaborative Web agent, Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'96), 1996, pp. 273–274.

[40] K. Sycara, Multi agent systems, AI Magazine 19 (2) (1998) 79–92.

[41] K. Sycara, D. Zeng, Coordination of multiple intelligent software agents, International Journal of Cooperative Information System 5 (2&3) (1996) 181–211.

[42] K.M. Tolle, H. Chen, Comparing noun phrasing techniques for use with medical digital library tools, Journal of the American Society for Information Science 51 (4) (2000) 352–370.

[43] C.J. van Rijsbergen, Information Retrieval, 2nd edn., Butterworth, London, 1979.

[44] A. Veerasamy, N.J. Belkin, Evaluation of a tool for visualization of information retrieval results, Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'96), 1996, pp. 85–92.

[45] E. Voorhees, D. Harman, Overview of the sixth text retrieval conference (TREC-6), in: E. Voorhees, D. Harman (Eds.), NIST Special Publication 500-240: The Sixth Text Retrieval Conference (TREC-6), Gaithersburg, MD, USA, 1997.

[46] A.M.A. Wasfi, Collecting user access patterns for building user profiles and collaborative filtering, Proceedings of the 1999 International Conference on Intelligent User Interfaces (IUI'99), 1999, pp. 57–64.

[47] C. Yang, J. Yen, H. Chen, Intelligent internet searching agent based on hybrid simulated annealing, Decision Support Systems 28 (2000) 269–277.

[48] O. Zamir, O. Etzioni, Grouper: a dynamic clustering interface to web search results, Proceedings of the 8th International World Wide Web Conference (WWW8), Toronto, Canada, May 1999.

Michael C. Chau is a doctoral student in the Department of Management Information Systems at the University of Arizona, where he is also a research associate of the Artificial Intelligence Lab. His current research interests include information retrieval, natural language processing, Web mining, and multi-agent systems. He received a B.S. in Computer Science (Information Systems) from the University of Hong Kong.

Daniel Dajun Zeng is an assistant professor in the Department of Management Information Systems at the University of Arizona. His research interests include software agents and their applications, distributed artificial intelligence, distributed decision support systems, negotiation, multi-agent learning, supply chain management, and intelligent information gathering. He received MS and PhD degrees in industrial administration from Carnegie Mellon University, and a B.S. in economics and operations research from the University of Science and Technology of China, Hefei, China. He is a member of INFORMS and AAAI.

Hsinchun Chen is McClelland Professor of MIS and Andersen Professor of MIS at the University of Arizona, where he is the director of the Artificial Intelligence Lab and the director of the Hoffman E-Commerce Lab. His articles have appeared in *Communications of the ACM*, *IEEE Computer*, *Journal of the American Society for Information Science and Technology*, *IEEE Expert*, and many other publications. Professor Chen has received grant awards from NSF, DARPA, NASA, NIH, NIJ, NLM, NCSA, HP, SAP, 3COM, and AT&T. He serves on the editorial board of *Decision Support Systems* and the *Journal of the American Society for Information Science and Technology*, and has served as the conference general chair of the International Conferences on Asian Digital Library in the past 4 years.

Michael Huang recently obtained his MS degree in Management Information Systems from the University of Arizona. He has been a research assistant at the Artificial Intelligence Lab for the past 2 years. His research interests are data warehousing, collaborative information retrieval, and knowledge management.

David Hendriawan received his Bachelor of Science in Business Administration from Parahyangan Catholic University, Indonesia, in 1996. He was a master's student in Management Information Systems at the University of Arizona and worked as a research assistant at the Artificial Intelligence Lab. He graduated in 2000. His research interests include information retrieval and multi-agent systems.