The Research on Multi-Agent Intelligent Information Retrieval System

Yan Li
College of Computer and Communication Engineering, Weifang University, Weifang, China
Email: liyan@wfu.edu.cn

Abstract—In the domain of multi-user and agent-oriented information systems, personalized information systems aim to give specific and customized responses to individual user requests. According to the current status of information search, a new personalized information search model is reported and the design and implementation of a system based on intelligent multi-agent is presented. A personalized information model must specialize to current interests of the user and adapt as they change over time. This paper builds a user personalized model to aid the accuracy of information distilling in order to enhance the personalized, and several models of agent interaction has been proposed with the aim of creating a multi-agent system for information personalization.

Index Terms—multi-agent system, information retrieval, Information Filtering

I. INTRODUCTION

The number of networked users has increased rapidly with the widespread proliferation of computers and networks. Information has become an instrument that can be used for solving problems. The Internet has vast potential for further development, new applications, and new solutions. One of the challenging problems that needs an improved solution is the problem of Internet Information Retrieval. Information retrieval research involves techniques from machine learning and other theoretical models, together with extensive experimentation to develop more accurate, fast and advanced information retrieval and search techniques for a variety of applications [1]. A tremendous amount of news and information is created and delivered over electronic media. This has made it increasingly difficult for individuals to control and effectively manage the potentially infinite flow of information. Ironically, just as more and more users are getting online, it is getting increasingly difficult to find information unless one knows exactly where to get it from and how to get it. Tools to regulate the flow are urgently needed to prevent computer users from being drowned by the flood of incoming information. Most present information retrieving tools have flaws as follows[2]:

(1) Low quality of web page capturing. Collecting a large body of web pages easily leads to low quality and information redundancy.

(2) Low precision ratio.

(3) Lack of personal services.

The agent domain can provide the means for adapting results to system users. Agent Paradigm is a promising technology for information retrieval. An agent-based approach means that IR systems can be more scalable, flexible, extensible, and interoperable. Agent originates in distributed artificial intelligence field (DAI), and it is a term of artificial intelligence. As an abstract entity, Agent can act on itself as well as the environment, and it also can react to the environment [3]. The idea of employing agents to delegate computer-based tasks goes back to research by Kay[4]. The research in this field is directed towards the ideal of agents that have high-level goals human-like communication skills and can accept high-level goals and reliably translate these to low level tasks. In DAI, the researcher endows different structures, contents and abilities on his own system in order to make it convenient for his specific research direction. Generally, Agent should have knowledge, objective and ability.

An agent specification framework must be capable of capturing the aspects of beliefs that agents have; ongoing interaction that agents have with their environment; goals that agents will try to achieve; and actions that agents perform and the effects of these actions [5].

Multi-agent system (MAS) are the emerging subfield of AI that aims to provide both principles for construction of complex systems involving multiple agents and mechanisms for coordination of independent agents’ behaviors [6]. Agents in a MAS are normally more independent units, that may have to compete for the use of resources, and since they do not necessarily use the same language, they must address the problem of translating other agents functions and mapping their behavior into their own individual representation [7]. In a society of agents a crucial issue is how agents exchange information i.e. communication. Communication involves agent “conversations” according to some coordination schema in order to achieve the desired co-operative problem solving. These conversations require the definition of two independent layers: the communication and the interaction layer. The communication layer specifies how the information is transmitted, the type of messages exchanged and the ontology for interpreting the content of the messages, i.e. an Agent Communication Language (ACL) such as KQML. The interaction defines how agents cooperate, the way that are engaged in conversations to divide the work, share information or solve potential conflicts between them. Interaction models include a number of techniques such as contract nets, multi-agent planning and negotiation.

The remainder of this paper is organized as followings: Section 2 introduces system framework and its
components function. Section 3 introduces feedback for retrieved documents. Section 4 introduces the results of experiment. Section 5 summarizes the paper.

II. SYSTEM DESIGN

A. System Framework

In this section, we will briefly describe the design of our framework. System framework is shown in figure 1. There are 6 main components in this system.

![Multi-Agent information retrieval System framework](image)

**Figure 1. Multi-Agent information retrieval System framework**

B. The Functionality of Each Agent

The functionality of each Agent are described as below:

1. **User personalized model.** This model is used to keep user's interface style [8], user’s inputting keywords each time, professional glossary which represents user’s background knowledge, the operation manner in specific circumstance, subject terms of web pages which user download each time. The styles keep track of the relationships between the agent and its neighbors, updating the neighborhood parameters.

2. **Information resource base.** This base is used to store obtained web pages information, dictionary, and training documents. The dictionary listing the concepts that the agent knows. The dictionary allows the agent to compare and discover relationships between a pair of words or concepts. Each concept has a list of supporting documents or links.

3. **Interface Agent.** User interface is an important component of application software. Interface Agent supplies user a friendly operating interface with features of individualism and intelligentization for their inputting requirements [9]. Requirement can be input through the key board. The Agent uses guiding information to help user determine the domain his/her requiring information belongs to, supplies specification and regulation to make search more precise. It is also able to accept the results from retrieval Agent and display them to the user for him to evaluate the feedback, then learn the user's feedback information, and dynamically modify and perfect user's personalized model.

4. **Search Agent.** Its function is to realize personal and intelligent searching according to user’s interest and requirements. It is designed to download records from the internet. Those records serve as the training set to create an entry vocabulary module. Search Agent gets user’s searching requirements from Retrieve Agent and subjects from use personalized model, carries evidence which is used to judge if web pages are in accordance those requirements and subjects so as to realize collecting and downloading intelligently. At last, Agent adds those correspondent pages into information resource base.

5. **Retrieve Agent.** It is designed to receive the specification request from Interface Agent and Send retrieving results back to interface Agent. If request exists in use personalized model, then search information resource base directly. Otherwise, sends user’s request to search Agent.

6. **Information filter Agent.** Information filtering deals with the delivery of information that is relevant to the user in a timely manner. This Agent filters and selects web pages obtained by search Agent. Also, it classifies pages according to certain regulation and save them into information resource base.

The focus of the current phase of our research is on developing and analyzing the operational components of our framework—how the agents interact and learn from each other—applied to an information retrieval problem. Interface agent serves a user who submits queries based on keywords. These keywords are known as concepts in the agents. The goal of this problem is satisfy as many queries as possible and as well as possible.

III. INFORMATION FILTERING AND FEEDBACK

A. Information Filtering

An intelligent information filter [10] means that it can analyze text or other entities automatically according to certain standard and determines their classification. Information filter agent assists the user with the task of finding interesting documents obtained by search Agent. It has technical knowledge about the task involved, namely, information filtering. It is also aware of the interests and preferences of the user. Agent uses this knowledge to automate filtering tasks for the user. A vector space model [11] was proposed to deal with documents on a computer. In a VSM, a document and a user profile are described as a vector. Hence, the similarity between the document and the user profile is defined as an inner product. When a request is received, it is translated into its vector representation and document vectors in the proximity of the request vector are retrieved in response to the search. The advantage of using a common vector space for both documents and requests is that document can also be used as a request itself. i.e. one can find documents that are similar to a given document. A standard method of indexing document consists of recognizing individual words, eliminating the commonly used words incuded on a word –exclusion list and using the remaining words for content identification of the document. We now assume that A
document vector is a row vector whose elements are the weights of words in a text. The i-th text vector di is denoted as

$$d_i = [w_{i1}, \ldots, w_{im}]$$  (1)

where \(w_{ij}\) means the weight for the j-th word in the texts, m is the number of words. If the j-th word is included in interesting text, the value of \(w_{ij}\) becomes high. On the other hand, if the j-th word exists in uninteresting documents, the value of \(w_{ij}\) becomes low.

A document consists of many fields. The text is just one of the many fields. The other fields could include author, location( the geographic origin of the news article), date( when the article was posted) etc. There is no restriction on the number of fields, so long as it represents some attribute of the document. Each field is assigned words to be used for identifying purpose. Since the words are not all equally important, they are assigned weight. The i-th field vector \(d^f_i\) is denoted as

$$d^f_i = [w^f_{i1}, w^f_{i2}, \ldots, w^f_{ij}, \ldots, w^f_{im}]$$  (2)

where \(w^f_{ij}\) is the weight of word \(t^f_j\) in the field \(d^f_i\). The superscript \(f\) indicates that \(d^f_i\) is a document field.

Since a document consists of many fields, it is represented as a set of field-vectors. Formally,

$$D = [d^f_1, \ldots, d^f_l, \ldots, d^f_l]$$  (3)

where \(d^f_i\) is the i-th field in document D.

In order for an information filter agent to assess its similarity to a given WWW page, it has to compare its item to the vector representation of the text inside that page. As mentioned above, each document is represented by a multi-dimensional vector each vector can have different dimensions. When two documents vectors are compared for similarity, we compute the cosine of the angle between the two vectors. This is done by evaluating the dot product of the two vectors and dividing it by the product of their magnitudes. The formula that returns the distance between two documents vectors \(d_i\) and \(d_j\) is

$$Sim(d_i, d_j) = \cos \theta = \frac{\sum_{l=1}^{m} w^f_{il} w^f_{lj}}{\sqrt{\sum_{l=1}^{m} w^f_{il}^2} \sqrt{\sum_{l=1}^{m} w^f_{lj}^2}}$$  (4)

B. Feedback for retrieved documents

Relevance feedback has been used to improve the performance of retrieval systems [12]. For vector space representations, the method for request reformulation in response to user feedback is vector adjustment. Since requests and documents are both vectors, the request vector is moved closer to the vectors representing documents which received positive feedback and away from the vectors representing documents which received negative feedback. Each of the field-vectors in the profile is modified in response to user feedback. Consider a profile \(P\), which contributed document \(D\) for presentation to the user. The user provides feedback \(f\), which is a positive or negative integer indicating the amount of feedback. Each field-vector in the profile is changed in proportion to the feedback received,

$$P = P + f \times D \times \delta$$  (5)

where \(\delta\) is the learning rate which indicates the sensitivity of the profile to user feedback.

Addition in the context of equation (5) means

$$w^p_{ik} = w^p_{ik} + f \times w^f_{ik} \times \delta$$  (6)

where \(w^p_{ik}\) is the weight of term \(t_k\) in field \(i\) of the profile and \(w^f_{ik}\) is the weight of the same term in field \(i\) of the document. The resulting effect is that, for those terms already present in the profile, the term-weights are modified in proportion to the feedback. Terms not already in the profile are added to the profile.

IV. EXPERIMENTAL RESULTS

Information retrieval system has plenty of data to be revealed to users, so select the Applet as the client program to supply a good data view based on graphics interface, due to the powerful functions and profuse graphic components of Applet in Java [13]. In the experiments, we concentrate on evaluating the learning ability of interface agent. We first investigate how a preference model is evolved by the learning mechanism, and then conduct a series of experiments to test the model.

We calculated ten users’ retrieving results by using commonly retrieving system and our system. The results were analyzed by general appraising method which has two indicators: recall ratio and precision ratio. Result is shown in table 1. The experimentation results show that our retrieving system get the better recall ratio and precision ratio than that of traditional system obviously.

<table>
<thead>
<tr>
<th>TABLE I. EXPERIMENTAL RESULTS</th>
<th>Appraiser Indicator</th>
<th>Commonly Retrieving System</th>
<th>Multi-Agent Retrieval System</th>
</tr>
</thead>
<tbody>
<tr>
<td>recall ratio</td>
<td>27%</td>
<td>56%</td>
<td></td>
</tr>
<tr>
<td>precision ratio</td>
<td>43%</td>
<td>67%</td>
<td></td>
</tr>
</tbody>
</table>

V. CONCLUSION

This paper combines traditional retrieving technology and Multi-Agent technology, propose an intelligent multi-agent information retrieval system. Model’s architecture and work principle have been analyzed and solution to system’s critical problems have been given. Furthermore, the system can learn to fit for different users’ manners and need to satisfy the individualized need. Further research will refine the search model to make retrieval efficient.
ACKNOWLEDGMENT

This work was supported by the Shandong scientific and technological project (Grant Nos. 2008GG30001030), and Shandong Natural Science Foundation (Nos. Y2007G65), and Shandong scientific and technological project (Grant Nos. 2009RKA545).

REFERENCES