AUTONOMOUS AGENTS FOR WEB PAGES FILTERING

FLÁVIA COIMBRA DELICATO, LUCI PIRMEZ AND LUIZ FERNANDO RUST DA COSTA CARMO

Núcleo de Computação Eletrônica, Federal University of Rio de Janeiro, Epitacio Pessoa 4476, 803 – Bl. 1, Lagoa, Rio de Janeiro, RJ
BRAZIL
E-mails: flavia@hotmail.com; luci@nce.ufrj.br; rust@nce.ufrj.br

With the current growth of the information available in Internet, users are facing an information overload. This work proposes a multiagent system for Web pages personalized filtering. The system is composed of a set of autonomous and adaptive agents that automatically provide relevant documents to the user according to a preferences profile. The agents learn with the user feedback and attempt to produce better results over time. This work presents the system description and the promising results of tests performed in a simulated environment. The proposed system proved to be a useful tool to recommend successfully relevant information to a well-defined preferences user.

1 Introduction

The use of the Internet has been growing in the last years with the appearance of World Wide Web. Although the increase of the information available facilitates the spreading of knowledge and the acquisition of products and services, it also makes the search for relevant material a real challenge. Recent work that arises at the intersection on information retrieval and software agents offers some new solutions to this problem. Agents can be defined as softwares with the aim of performing tasks for their users, usually with autonomy, playing the role of personal assistants.

The present work suggests the use of autonomous agents for the personalized information filtering. The proposed system is composed of a set of adaptive and non-mobile agents aiming to satisfy the user's needs for information. The agents receive the user's feedback about the relevance of the retrieved information and improve their search, obtaining better results over time.

The set of agents is autonomous as it can perform its task without the user's presence, based on a preference profile previously built. The system is adaptive as it learns the user's preferences and adapts itself when these ones change over time. The main agent's learning mechanisms is the relevance feedback [7]. The use of

genetic algorithms [4] as a complementary mechanism aiming to introduce diversity in the system's parameters is addressed. The information is represented by the vector space model [8]. The results presented were obtained through a series of sessions with simulated users. The system's efficiency evaluation was made through the normalized distance performance measure (ndpm), suggested by Yao [9].

This paper is organized as follows. In section 2 there's a comparison with related works. Section 3 describes the proposed system. The analysis of results is presented in section 4 and some conclusions are drawn in section 5.

2 Related Works

In the domain of Web, WebWatcher [1] and Lira[2] are agents whose actions are interleaved with the user's browsing in Netscape. They require explicit interaction to indicate interest in topics or particular pages. MIT Media Laboratory's Letizia [5] is an autonomous interface agent designed to assist and provide personalization to the user while browsing the WWW by performing a breadth-first search on the links ahead and providing navegation recommendations. The main disadvantage of such approaches is that they are restricted to the sections of the Web visited by the user, recommending links starting from them. In contrast, the proposed system looks for new domains for information that can be of potential interest for the user. The user probably never saw before the presented topic.

More similar to our work with regards to application domain and representation are the systems built by Balabanovic [3] and Amalthea, proposed by Moukas [6]. Balabanovic proposed a multiagent system that combines both content-based and collaborative techniques applied to the web pages recommendation. He adopts the vector-space model, relevance feedback as the learning method and he suggests the use of genetic algorithms as a possible solution for some of the problems found in the content-based filtering. Amalthea is a system that combines the concepts of autonomous agents and artificial life in the creation of an evolving ecosystem composed of competing and cooperating agents for web pages recommendation.

3 System Description

Fenix system was developed according to the object oriented approach. The system was implemented as a Java application to be running locally in the user's machine. Fenix is composed of various functional modules described below.

3.1 User Interface Module

This module presents a graphic interface to interact with the user. The user's interaction with Fenix system begins with his registration, where he must inform his

personal data and choose a login and a password. After the identification the user can create a new agent, load an existing one or to activate the autonomous mode.

When creating a new agent, the user must choose a name and provide the search parameters, which include the query expression. As a result of the initial search, a series of retrieved documents is presented. After reading the chosen documents, the user can provide positive (+1) or negative (-1) feedback according to their relevance. The user can modify a document URL, if he finds a more interesting link starting from the initial page, through the button "Alters". He can also include a URL of interest manually, that has not been retrieved by the agent, by clicking the button “Includes”.

When saving a newly created agent, the references to the documents with positive feedback will be saved (their URLs) and the term vector and their weights will be created, building the initial profiles for that agent. The user can also choose some URLs to be constantly monitored. Certain URLs are frequently changed and updated, and the system can be scheduled to verify from time to time if their contents changed.

When loading an existing agent, the user can read some retrieved document, provide feedback about some document or starting a new search for documents.

3.2 Filtering Module

The filtering process consists in translating documents to their vector representations, calculate the similarity between documents and profiles, and selecting the top-scoring documents for presentation to the user.

The representation adopted in this work is based on the vector space model (VSM) [8]. To compute the content of a document, the system uses a keyword frequency measure, TFIDF (term frequency times inverse document frequency). This technique says that keywords that are relatively common in the document, but relatively rare in general are good indicators of the content [7].

A profile is a set of information about the retrieved documents as, for example, the documents URL, the score computed for the system and the user's feedback assigned to them. Besides, it contains the vector representation of all documents that received positive feedback. The adopted similarity measure was the cosine of the angle between the documents and the profiles vectors [8].

The filtering agents are responsible for gathering the documents generated by all the profiles, classifying them according to their similarity values, eliminating repetitions and presenting to the user.

3.3 Learning Module

The learning methods addressed in this work were relevance feedback and genetic algorithms. At the present stage, the relevance feedback sub-module was

implemented and tested. The specifications of the genetic algorithm sub-module had already been done, but its implementation will be a matter of future works.

- *Relevance Feedback Sub-Module*

For vector space representations, the method for query reformulation in response to user's feedback is vector adjustment. Since queries and documents are both vectors, the query vector is moved closer to vector representing documents with positive feedback, and further from vectors of the documents with negative feedback. The effect is that, for those terms already existing in the profile, the term weights are modified in proportion to the feedback. The terms not existing in the profile must be added to it.

- *Genetic Algorithm Sub-Module*

In the next stage of this work the genetic algorithm will be implemented as a complementary mechanism to introduce diversity to the search parameters as a goal. This goal will be achieved by recombining the contents of different vectors of terms belonging to the same user profile.

In the proposed system, a population P is defined as a group, where each element is a pair of profile and its fittness. Each profile is converted for a binary representation, and it corresponds to an individual or chromosome of the population. The fittness is computed based on the average values of similarity between the documents and their respective profiles. The genetic operators of crossover and mutation update the population to each generation, introducing new members and taking advantage of the fittest ones. The final objective is to evolve the population in direction to a global optimization.

3.4 Other Modules

The search module is responsible for interacting with search engines existent in the web, gathering information from them about the chosen subject and saving them in a local database. The system database is composed of all information from the user, his agents and respective profiles, as well as the pages retrieved in searches. Besides these modules, Fenix has a controlling module, responsible for controlling all the other classes creation and their methods invocation.

4 Results

We adopted the performance measure proposed by Yao [9]. The ndpm measure ("normalized distance-based performance measure") is a distance, normalized to range from 0 to 1, between the user's classification for a set of documents and the system's classification for the same documents. A user is supplied with a list of

documents and should classify it in agreement with his interests by a subject. The system also ranks the documents according to how well they match the profile previously built for that user. The expected result is for the ndpm distance between the user and system classifications to decrease gradually over time, as the user's profile is adjusted.

One hundred and twenty agents were created for thirty subjects of interest to a simulated user. For each subject, a number of simulated sessions of "user"-system interaction were accomplished. After an initial search, the agents classified the retrieved documents; the "user" evaluated the documents, providing their feedback values and classification. With the feedback, the agents profiles were adjusted to further searches and the classification is used to computer the ndpm.

A progressive decrease of the ndpm distance along the sessions was observed, indicating that the agents were adapting themselves to the user's preferences and increasing the probability of retrieving a larger number of relevant documents while discarding the irrelevant ones.

Several system configuration parameters were tested in the simulated sessions. To sum up the final results we can say the system reached the best performance when the terms of the query were more specific, the agents were composed of at least 4 (four) and in the maximum 10 (ten) profiles; and the term vectors of the documents had maximum size of 300 terms.

Nine agents were tested along 20 (twenty) sessions, in order to compare with the work described in [3], where a multiagent system was implemented for the WWW pages recommendation. His system performance was also evaluated with the ndpm measure and the obtained curve had a behavior quite similar to the one presented in the tests with Fenix (Figure 1).

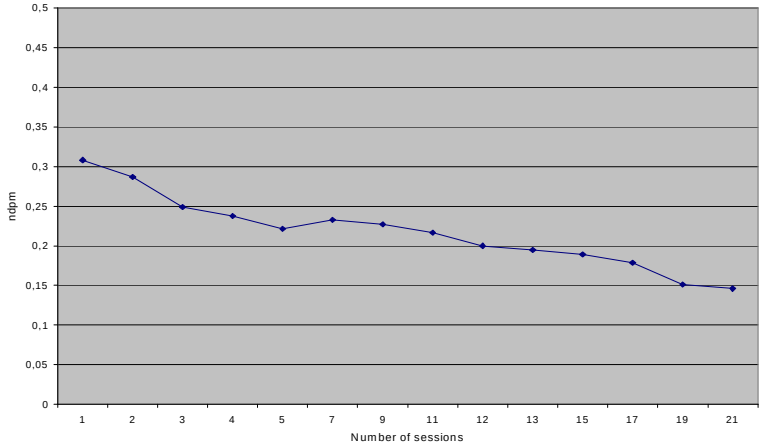

Figure 1: Average ndpm distance between user and system rankings, over 20 sessions.

5 CONCLUSIONS

Fenix is an autonomous agent that must be able to specialize to user interests, to adapt when they change and to explore the domain for potentially relevant information. The system proved to be a powerful tool of information filtering. The presented results confirmed that the system, using the vector-space model with relevance feedback as the learning mechanism is able to successfully filter relevant documents for a well-defined preferences user.

The performance values obtained in simulated tests based on the ndpm measure were similar to the ones found in another works of information filtering.

Information filtering agents are a great promise to the management of extensive available information.

REFERENCES

1. Armstrong, R. et all., WebWatcher: A Learning Apprentice for the World Wide Web, in AAAI Spring Symposium on Information Gathering, Stanford, CA, March 1995. Available in: http://www.cs.cmu.edu/afs/cs/project/theo-6/web-agent/www/project-home.html.
2. Balabanovic, M. and Shoham, Y., Learning Information Retrieval Agents: Experiments with Automated Web Browsing, , in AAAI Spring Symposium on Information Gathering, Stanford, CA, March 1995. Available in: http://flamingo.stanford.edu/ users/ marko/ bio.html.
3. Balabanovic , M. Learning to Surf: Multiagent Systems for Adaptive Web Page Recomendation Service. Dissertation submitted to the Department of Computer Science and the Committee on Graduate Studies of Stanford University. UMI Number: 9837173. UMI Company. 1998.
4. Goldberg, D. E. Genetic and Evolutionary Algorithms come of age. Communications of the ACM, 37(3):113-119, March 1994.
5. Lieberman, H., Letizia, an agent that assists web browsing. In *Proceedings of IJCAI-95*. AAAI Press, 1995.
6. Moukas, A., *Amalthaea:* Information Discovery and Filtering using a Multiagent Evolving Ecosystem. In proceedings of the Conference on Practical Applications of Agents and Multiagent Technology, London, April 1996
7. Rocchio, J.J. Relevance feedback in information retrieval**.** In: The Smart Retrieval System - Experiments in automatic Document Processing, p. 313-323, Englewood Cliffs: Prentice-Hall, 1971.
8. Salton, G., Automatic Text Processing – The Transformation, Analysis and Retrieval of Information by Computer. Addison-Wesley Publishing Company, Inc., Reading, MA, 1989.
9. Yao, Y. Y. 1995. Measuring retrieval effectiveness based on user preference of documents. Journal of the American Society for Information Science 46(2):133-145.