IWFS - An Intelligent Web Filtering System

Abdelhakim Herrouz[1], Mahieddine Djoudi[2]
[1]Department of Computer Science,
University Kasdi Merbah of Ouargla, Algeria
abdelhakim.herrouz@gmail.com
[2]Department XLIM-SIC UMR CNRS 7252 & TechNE Research Group,
University of Poitiers
Téléport 2, Boulevard Marie et Pierre Curie, BP 30179
86960 Futuroscope Cedex, France
mahieddine.djoudi@univ-poitiers.fr

The excessive information on the Web creates the information overflow problem. This work suggests the use of intelligent agents for the personalized filtering of Web pages. Several agents are designed and implemented, including a Web page reader agent and an intelligent agent that assists a user by filtering information. The user can specify a profile of keywords and provide explicit ratings for each article. Two alternate methods are provided for filtering the articles, including keyword match and clustering. This paper presents the system description and the promising results of tests performed in a real environment. The proposed system has proven to be a useful tool in reducing the amount of information the user has to deal with.

Keywords: Agent-based System, Agents, Web Pages, User Profile, Personalized Information Filtering, Neural Networks

1. INTRODUCTION

Due to the rapid growth of the Web, sites appear and disappear, content is modified and it becomes impossible to master their organization. The nature of the environment itself imposes some disadvantages: Internet is a network of worldwide level, constantly changing and non-structured[1]. The Web is the largest data source in the world and it has become one of the most widespread platforms for information change and retrieval. As it becomes easier to publish documents, as the number of users, and thus publishers, increases and as the number of documents grows, searching for information is turning into a cumbersome and time-consuming operation[2].

When a user wants to find interesting documents or date sources, the user has to actively search the World Wide Web. Searchers required effective means to efficiently find the information that they really need, and avoid the irrelevant information that does not match their interests. Information retrieval, and information filtering are two major information access techniques.

Information retrieval is concerned with retrieving the relevant documents from a large collection of material efficiently. It is concerned with the collection, representation, storage, organization, access, manipulation and display of information.

The immense volume of source information, however, often leads to query results which are too long and unwieldy for human users to manage effectively. The need, therefore, arises for more intelligent aids for information access tasks. Information filtering is an example of such an information access process.

Unlike information retrieval, information filtering generally focuses on users' long-term information needs, often being stable preferences. It operates on dynamically changing information streams (e.g. email and news). Based on a user's prole, which is initially derived from his or her interests, a filtering system processes a new item and takes appropriate actions that either ignore it or bring it to the user's notice[3].

This paper is organized as follows: In section 2 there is a presentation of Information Filtering Systems. Section 3 presents the learning systems. The software architecture is described in section 4. The experiments and interpretation of results are presented in section 5 and, finally, some conclusions are drawn in section 6.

2. INFORMATION FILTERING SYSTEMS

Information Filtering (IF) has begun to attract attention as a method for delivery of relevant information. IF systems cover a broad range of domains, technologies and methods involved in the process of exposing users to the information they need. IF systems:

- are applicable for unstructured or semi-structured data (e.g. documents, e-mail messages);
- handle large amounts of data;
- deal primarily with textual data;
- are based on user profiles; and
- their objective is to remove irrelevant data from incoming streams of data items.

Many of the above features are not exclusive to IF systems and can be found in other text-based information systems such as information retrieval (IR), extraction and categorization systems[4].

FILTERING TYPES

A. Content-based

Content Based filtering system recommends a document by matching the document profile with the user profile, using traditional information retrieval techniques such Term Frequency and Inverse Document frequency (TF-IDF). User characteristics are gathered over time and profiled automatically based upon a user's prior feedback and choices. The system uses item to item correlation in recommending the document to the user. The system starts with the process of collecting the content details about the item, such as treatments, symptoms etc. for disease related item and author, publisher etc. for the book items. In the next step, the system asks the user to rate the items. Finally, system matches unrated item with the user profile item and assign score to the unrated item and user is presented with items ranked according to the scores assigned[5].

B. Collaborative Filtering

Collaborative filtering systems filters information based on the interests of the user (past history), and the ratings of other users with similar interests. It is widely used in many filtering systems or recommender systems, especially in ecommerce applications. One of the examples of such system are Amazon.com and e-Bay, where a user's past shopping history is used to make recommendations for new products.

C. Hybrid Filtering Systems

The hybrid filtering systems combines features of both the content and collaborative filtering systems. The hybrid system overcomes the problem of cold start and early rater problem by using the content based approach in the initial stage. In the subsequent stages, it uses collaborative filtering systems features, which helps the system to recommend all types of items, including multimedia items and overcomes the problem related to content based filtering techniques.

3. LEARNING SYSTEMS

3.1 LEARNING AGENTS

The central element of intelligence behavior is the ability to adapt or learn from experience. There is no way that we can know a priori all of the situations that our system will encounter. Any agent that can learn has an advantage over one that cannot. Adding learning or adaptive behavior to an intelligent agent elevates it to a higher level of ability. A learning agent can adapt to our likes and dislikes. It can learn which agents to trust and cooperate with, and which ones to avoid. A learning agent can recognize situations it has been in before and improve its performance based on prior experience when it encounters the same situation again[6].

One form of learning is called clustering. It looks at high-dimensional data –data with many attributes- and scores them for similarity based on some criterion. The clustering technique is used in data mining tools. Data mining is commonly defined as the process of discovering useful patterns or knowledge from data sources (e.g., databases, texts, images, the Web, etc)[2]. The main contribution of data mining is to find patterns which were not known to exist, that is, to discover new information or knowledge. So learning, as applied to data mining, can be thought of as a way for intelligent agents to automatically discover knowledge rather than having it predefined using predicate logic, rules or some other knowledge representation[6].

The unsupervised learning is a paradigm used when the learning agent needs to recognize similarities between inputs or to identify features in the input data. The data is presented to the agent, and it adapts so that it partitions the data into groups. The clustering or segmentation process continues until the agent places the same data into the same group on successive passes over the data. Our IWFS system uses a neural network implementation of unsupervised learning technique called Kohonen Map.

An important distinction in learning agents is whether the learning is done on-line or off-line:

- On-line learning: means that the agent is sent out to perform its tasks and it can learn or adapt after each transaction is processed. It must be very fast and very stable.

- Off-line learning: means that we would gather data from situations that system has experienced. We could then augment this data with information about the desired agent response to build a training data set.

Once we have the database, we can use it to modify the behavior of our agents.

3.2 KOHONEN MAP NEURAL NETWORK

A Neural Network is an interconnected group of natural or artificial neurons that uses a mathematical or computational model for information processing based on a connections approach to computation. In most cases an Artificial Neural Network is an adaptive system that changes its structure based on external or internal information that flows through the network[7]. Neural networks provide an easy way to add learning ability to agents.

The Self-Organizing feature maps developed by T. Kohonen have become one of the most popular and practical neural network models. Kohonen is efficient tool in the important domain which is the unsupervised classification[8]. It is considered as a useful tool in data mining because it accelerates the research of relevant information. An unsupervised classifier groups the similar information that refer to the same topic in same cluster and these which are dissimilar in the distinct ones. This avoid the search of the desired information in a lot of clusters, consequently an important time is economized. The Kohonen algorithm is unsupervised partitioned classifier; it treats with unlabeled inputs and provides classes with no overlap. Beside its ability to resist to the noise, the Kohonen algorithm possesses other interesting properties. Indeed, the self-organizing map is an unsupervised neural network which projects high-dimensional data onto a low-dimensional grid which called a topological map[9]. A Kohonen Maps is a single-layer neural network, comprised of an input layer and an output layer (fig.1), and the self-organizing feature maps perform unsupervised learning:

- Each time an input vector is presented to the network, its distance (Euclidean distance) to each unit in the output layer is computed.
- The output unit with the smallest distance to the input vector is declared the "winner". This winning neuron is adapted by a non-supervised learning algorithm as described by Kohonen[10,11,9].
- The winning unit and a set of units in the neighborhood weights are adjusted by moving the weights toward the input vector.
- The Kohonen map self-organizes over time, and at the end of a successful training run, a topographic map is created.
- The distance of the input pattern to the weights to each output unit is computed using the Euclidean distance formula:

$$y_j = ||x - w_j||^2$$

where x is the input vector, w_j is the weight vector into output unit j, and y_j is the resulting distance.

- The weights of the winner and the units in its neighborhood are then adjusted using:

$$w_j(t+1) = w_j(t) + \alpha(k)\, C_{ij}(k)\, y_j(t)$$

where $\alpha(k)$ is the learn rate at iteration k, $C_{ij}(k)$ is the value of the neighborhood function for units i and j at iteration k.

- This neighborhood function $C_{ij}(k)$ is called a Gaussian function and is defined as:

$$C_{ij}(k) = \exp\left(\frac{-||i-j||^2}{\sigma(k)^2}\right)$$

where i and j are the coordinates of the units in the two-dimensional map, and k is the iteration number, $\sigma(k)^2$ is the width of the neighborhood function.

- The $\alpha(k)$ parameter is the learn rate for iteration k, it is computed as:

$$\alpha(k) = \beta_{initial}\left(\beta_{final} / \beta_{initial}\right)^{k/k_{max}}$$

where k_{max} term is the maximum number of iterations to be performed, and the learn rate $\alpha(k)$ exponentially decreases as the iteration number k gets larger. Typical values for $\beta_{initial}$ and β_{final} are 1.0 and 0.05 respectively.

These formulas are used as basis for our system implementation.

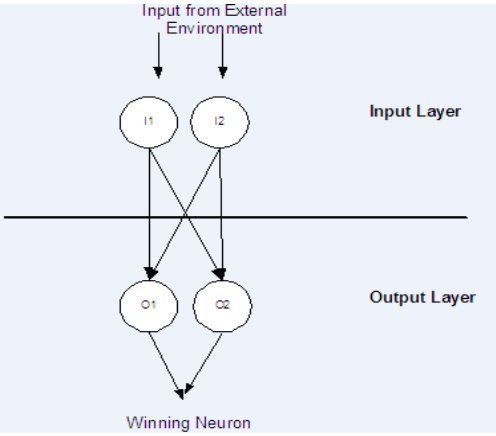


Fig.1. The structure of a typical Kohenen neural network.

4. SOFTWARE ARCHITECTURE

The application was developed using the Java programming language. The IWFS system allows Web page filtering based on two different filtering options. The primary agents used in the application consist of the URLReaderAgent and the FilterAgent (Fig. 2). The agents run autonomously to perform their actions. The URLReaderAgent is responsible for connecting to and downloading a page that is pointed to by a URL (*Uniform Resource Locator*). The FilterAgent uses two different types of filtering consisting of keyword filtering and cluster filtering. The keyword filter is used to track the number of occurrences of key words in the article that match the keywords that the user added to their profile. The clustering filter used a Kohonen map, a type of neural net where similar values are clustered together in the map, to group like topics together.

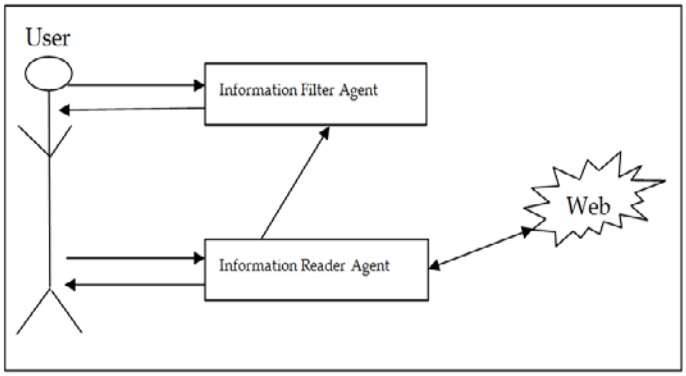
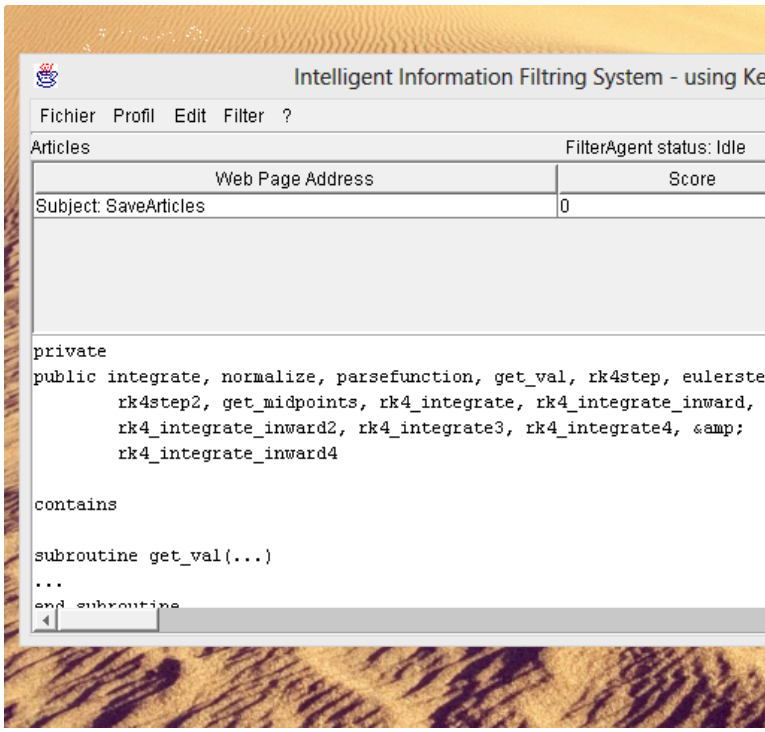


Fig.2. IWFS architecture

5. EXPERIMENTS



The web pages are loaded, ordered and grouped according to their qualification and level of interest for the user, by means of two filters:

- For the Keywords filter (by default), the qualification is simply the sum of all the keywords. The biggest the number of key words the more interesting it is.
- For the Cluster filter, the qualification is the average of the qualifications of the keywords of all the web pages that are inside the group. The higher the average of the group qualification the more interesting it is.

Before carrying out the classification it should be added the web pages of interest for training the neuronal network of the Cluster filter.

To test the IWFS application, in the first place, it was selected a group of keywords that will intervene in the classification of web pages:

N°	***Keywords***
1	Syntax
2	Memory
3	Language
4	Loop
5	Symbol
6	Reserved
7	Object
8	Public
9	Private
10	Program

Table.1. Keywords

Then, four web pages HTTP addresses were entered. The system qualified them with the Keywords filter, for default. Next, it recovers and stores the web pages and executes the phase of training of the neural net. The new qualification is calculated applying the Cluster filter, obtaining the following results:

KEYWORD FILTER

N°	*URL*	*Score*	*Rating*
1	http://www-control.eng.cam.ac.uk/~pcr20/C_Manual/chap10.html	50	Interesting
2	http://pymol.sourceforge.net/newman/user/S0210start_cmds.html	44	Interesting
3	http://www.extremeprogramming.org	1	Not very useful
4	http://fr.openclassrooms.com/forum/sujet/langage-fortran?page=1	1	Not very useful

Table.2. Classification using Keywords

INTERPRETATION

The web pages were classified by means of the Keywords Filter. In these cases, the pages were ordered by the quantity of keywords found in its content, including its repetitions.

CLUSTER FILTER

N°	*URL*	*Score*	*Rating*
1	http://www-control.eng.cam.ac.uk/~pcr20/C_Manual/chap10.html	47.0	Interesting
2	http://pymol.sourceforge.net/newman/user/S0210start_cmds.html	47.0	Interesting
3	http://www.extremeprogramming.org	1.0	Not very useful
4	http://fr.openclassrooms.com/forum/sujet/langage-fortran?page=1	1.0	Not very useful

Table.3. Classification using Cluster

INTERPRETATION

The Web pages were classified by means of the Cluster Filter. In these cases the Web pages form two groups, which are conformed according to the quantity of similar words that have.

As can be seen from the interpretations above, the page number 1 has the biggest number of keywords, although these are repeated. Once trained the neural network in the cluster method two groups are formed. The biggest qualification corresponds to the group formed by page number 1 and page number 2 (control.eng.cam.ac.uk and pymol.sourceforge.net). Finally, from the combined results it can be deduced that the web page that the system recommends is control.eng.cam.ac.uk.

5. CONCLUSIONS

We have developed a modest data-mining capability. The IWFS application provides the basic functionality of a general-purpose information filter. Agents are provided to gather source information from Internet Web pages. The URLReaderAgent reads the data from a single specified Web page (URL) and download it to the Personal Computer. After it is downloaded, each Web page is scored against a keyword list and optionally can be scored by using two neural networks. The user can rate each piece of information by using a five-level scale. The subject line from each URL is displayed in a table control. The user can browse individual Web pages sources by selecting them from the table.

One goal of the IWFS system is to show how an agent-based application can be used to Intelligently rank Web pages based on their content.

The FilterAgent can asynchronously trains the neural network models and we are in a position to reuse it in a composite agent or multi-agent system. However, with just a little additional work, the FilterAgent could be modified to build neural network models against any data set.

REFERENCES

[1] Herrouz, A., Djoudi, M. (2002). "Conception d'un Système d'Assistance à la Navigation et à l'Apprentissage sur Internet : SANA". Seventh Maghrebian Conference on Computer Sciences (7th MCSEAI), Annaba, Tome 1, pp 59-76., 6-8 mai.

[2] Herrouz A. Khentout, C. and Djoudi M., (2013). Overview of Web Content Mining Tools. The International Journal of Engineering and Science (IJES), Volume 2, Issue 6, ISSN: 2319 – 1813 ISBN: 2319 – 1805, 2013.

[3] Renganathan V, Babu AN, Sarbadhikari SN, (2013). A tutorial on Information Filtering Concepts and Methods for Bio-medical Searching. J. Health Med Informat 4.

[4] Hanani U., Shapira B. and Shoval P., (2001). Information Filtering: Overview of Issues, Research and Systems. User Modeling and User-Adapted Interaction 11: 203-259.

[5] Shareen, A., Nitin, S., (2015). A survey of Filtering System for OSN (Online Social Networks). IJCST – International journal of Computer Science and Technology – Vol. 6.

[6] Bigus J. P. and Bigus J., (2001). Constructing Intelligent Agents using Java (Professional Developer's Guide Series) / Edition 2, John Wiley & Sons, Inc.

[7] Rudolf V.., Igor C., (2013). Intelligent Communication Networks and Neural Networks. European International Journal of Science and Technology, vol.2, No.6, ISSN: 304-9693, pp. 29-40.

[8] Ettaouil M. et al., (2012). Learning Algorithm of Kohonen Network with Selection Phase. WSEAS TRANSACTIONS on COMPUTERS, Issue 11, Volume 11, November 2012, E-ISSN: 2224-2872, pp. 387-396.

[9] Ghorbel S., Ben Jemaa M. and Chtourou M., (2011). "Object-based Video compression using neural networks". IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No 1, July 2011.

[10] Kohonen T, (1988). An introduction to neural computing. Neural

Networks, Pergamon Journals Ltd, Vol. 1. pp. 3-16.
[11] Kohonen T, (1990). Self-Organisation Map. Proceedings of the IEEE, vol. 78, No. 9, pp. 1464-1480, September 1990.