An Evolutionary Algorithm to Optimize Web Document Retrieval

André L. Vizine[1], Leandro N. de Castro[1] & Ricardo R. Gudwin[2]
{vizine,lnunes}@unisantos.br; gudwin@dca.fee.unicamp.br
[1]Catholic University of Santos, R. Dr. Carvalho de Mendonça, 144, 11070-906, Santos/SP – Brazil;
[2]State University of Campinas, C.P. 6101, 13083-852, Campinas/SP – Brazil.

Abstract – *This paper presents an automatic keyword extraction method and an evolutionary algorithm that mines the web searching for documents according to group users interests. Both techniques were designed for future use in an academic virtual community, characterized as a scientific paper collection (PDF files) and a means for efficient knowledge and information exchange through the Web. The preliminary results presented here demonstrate that the parts of the system already implemented have a good potential for selecting appropriate libraries of keywords and, from them, making and optimizing queries for retrieving related documents from the Web.*

1. INTRODUCTION

The Internet can be seen as a global and distributed repository of resources and information. In most cases, these resources are immediately available for use and cover almost all domains, from the support of scientific and educative activities to recreation and entertainment. As a survey made by the UCLA Center Communication Policy, the three main reasons that make new people use or want to use the Internet are: to obtain and retrieve information quickly; professional needs; and communication (e.g., e-mail access) [1]. Moreover, the Internet is reducing the costs of production and distribution of information. As a result, an avalanche of material, in many cases of poor quality, is made available daily in the Web. Despite these benefits, the Internet is not adequately prepared for more abstract activities, such as the management, representation, and other types of information processing and exchange [2].

Along with the amount of information available, the number of people connected to the Internet and the number of web pages accessed have also increased exponentially over the past years. There is a great variety of resources and information available on the web for people with the most diverse background and interests. The major problems of the web, however, are that the bibliographical works available are spread all over the world, the speed with which this information is created and made available, and the poor quality of part of this information. It is thus, the readers' job to search for and filter out the relevant information. Even qualified users, such as academics (students, researchers and lecturers), do spend time searching and filtering the information retrieved from the Web.

Therefore, performing information filtering and flow efficiently becomes a necessary and challenging task. Information filtering systems are designed to filter out the information that a user requests from an enormous amount of information not always of interest [3]. The term information source is used here to represent the site where contents exist and are of interest to the user. These sources are often related to the places where a document collection exists in text form [4].

On one side, the technological advance makes it possible a network infrastructure that supports the most varied types of information resources (e.g., structured multimedia objects, documents and specialized data bases). On the other side, there is a need to develop client applications that assist the end user in the search, access, organization, and sharing of these information sources.

This paper describes a system to autonomously generate group profiles for web documents by selecting a suitable library of keywords, and a search agent that generates and optimizes, via a genetic algorithm (GA), search queries for the Google search engine. The libraries of the group profiles take into account the relative frequency of a word in a given document and its relative frequency in a set of related and unrelated documents [5]; an approach taken to insert context information into the system. The search agent uses a GA to optimize the search of new papers for a group of users instead of a single user. Both techniques were designed to be employed in an academic virtual community in a near future. This community will be characterized as a scientific paper collection (PDF files) automatically classified and stored in folder structures of a server and in which academics will be able to exchange experience and knowledge.

This article is structured as follows. Section 2 provides a brief overview of information filtering, representation of user profiles and web mining. Section 3 describes the method used for the construction of group profiles. Section 4 presents the genetic algorithm used for information filtering. Section 5 shows the performance evaluation of the algorithm and the work is concluded in Section 6 with a discussion about future avenues for investigation.

2. INFORMATION FILTERING

The goal of information filtering is to select (filter) information so as to quickly extract what is relevant to the user. However, the quality of the information varies according with the user. Information filtering systems will soon have to be personalized so as to serve particular interests, thus assuming the role of a personal assistant. A personalized information filtering system must satisfy three requisites [4]:

• **Specialization:** A personalized filtering system must serve the specific interests of the user. The amount of irrelevant texts delivered to the user must be as small as possible. The number of rejected relevant articles must also be small. The system must be able to identify standards of user behavior, infer its habits and adapt to them, make recommendations of relevant texts, and minimize the number of irrelevant recommendations.

• **Adaptation:** Since in the majority of times the user interests do not remain constant, when changes do occur, for instance, the interest for a new subject, the system has to be capable of perceiving and adapting to that change.

• **Exploration:** A filtering system must be capable of exploring new domains in order to find some novelties of potential interest to the user.

Belkin and Croft [3] provide a good description of information filtering and discuss some similarities and differences with the term information retrieval. Any process of information search starts with a description of the user interests. The distinct characteristics of the process of information filtering from information retrieval are that information filtering requires relatively specific information about the user interests, which tend to suffer modifications slowly with time. Information retrieval systems, by contrast, act on relatively steady sources of information to answer the queries made by the users. Therefore, the context of information filtering involves a set of dynamic information, as opposed to the static bases of the traditional systems of information retrieval. The main differences are summarized in Table I.

Table I – Information filtering × Information retrieval.

Process	Necessary Information	Resources of Information
Information Retrieval	Dynamic	Steady and structured
Information Filtering	Relatively Steady	Dynamic and unstructured

A personalized filtering system must readily take care of to the necessities of the user. The system must have the capacity to detect the user needs through an interactive process. Assuming that a major part of the actions taken by the user are relevant, the system will have to increase the quality of the suggestions (recommendations) made. Thus, the system will have to converge so that the user needs are consistently satisfied and sometimes foreseen.

The constant change of interest may be a simple diversification in data subjects, the interest for a completely new subject, or the loss of interest for a subject. The system must then be able to detect or to allow the user to indicate an interest change, and to adapt to this change. Finally, taking into account the motivation of the user in the search for information, we have to consider the hypothesis of the system being capable of recommending new and interesting sources of information based on the knowledge it has from the user.

2.1. User Profiles

Some systems have been developed using information filtering based on user profiles. The system SIFT [6] was designed for article filtering in the Internet through manually constructed user profiles. In this system, the user specifies which words are of interest and which are not. If the interests of the user change, these updates need to be manually incorporated into the profile. Another system developed, called InfoScope [7], also designed to deal with Internet News, deduced rules and presented them to the user waiting for his/her approval. These are extracted by the comments of the actions taken by the user, such as the time spent for reading the text or if the text was saved for future use. This prevented the need of an explicit feedback by the user on each text read. The employment of user profiles is so popular that in 1999 there was a workshop, WEBKDD [8], fully dedicated to user profiles in web mining.

2.2. Web Mining

The Internet is considered the largest library of the world [9]. Its major problem is that the "books" are spread all over the world without indexing. It is thus the readers' job to search the Web for a site where to find the desired contents. Furthermore, the reader still has to certify the quality of the information obtained. Although performing an exhaustive search on the Web is practically impossible, the use of a search engine solves part of the problem.

Search engines usually work by answering one or more queries entered by the user. This query (composed of Boolean keywords and operators) is matched with one or more databases, and the resources more similar to the query are returned. It is still the users' job to evaluate the quality of the information retrieved by the search engine in order to select the ones that are of greatest interest. The use of information filtering and data mining techniques in search engines is generically termed web mining [10].

3. GROUP PROFILES

In the last twenty years or so, research communities all over the world have been benefiting from the use of the Internet [10]-[13]. For instance, forty years ago it was very hard to share ideas with researchers in other conti-

nents and even researchers in the same country, but living in distant places. Bringing together a number of scientists for workshops, conferences and other types of meetings was a very hard and expensive task, sometimes unfeasible. Of course communication via Internet does not replace personal meetings, but it facilitates and reduces costs for the sharing of information.

According to [14], virtual communities can be defined as social aggregations that emerge on the net when a certain group of people makes large public discussions, thus forming a network of personal relationships on the web environment. Virtual communities allow the integration of different societies, the convergence of diversities and the production of common interests, thus approaching people. The Internet hosts a large number of communities with the most varied interests.

The present paper introduces an automatic keyword extraction method and a genetic algorithm to optimize web search. Both techniques were designed to be used in an academic virtual community characterized as a scientific paper collection (PDF files) automatically classified and stored in folder structures of a server and with the capability of exchanging information/knowledge among users. For a user to have access to these sources of information, he/she will have to login in one or more areas of interest. When a user (member of the community) finds an article interesting that is still not indexed in the community, this can be suggested for inclusion. With time, these folder structures start to increase in terms of number of documents and also in terms of quality of contents, since the papers are selected based on users having common interests and user evaluations. Every time the members of the community suggest a new paper, the group profile will be updated. Group profiles will be composed of a set of keywords extracted from the suggested papers.

3.1. Keyword Extraction

For a word *w* to represent a group profile; that is, to be selected as a keyword, it has to be a good descriptor of a group and represent a set of documents belonging to the group or folder *D*. The word *w* thus must have the following properties:

1. be predominant in *D* when compared with the other words in *D*;
2. be predominant in *D* when compared to its occurrence in all other sets of documents (folders).

The keyword selection method was taken from [15] and works as follows. Let $G(w)$ be the rank of a word *w*

$$G(w) = F^{cluster}(w) \times F^{coll}(w) \quad \textbf{(1)}$$

where $F^{cluster}$ relates word *w* with the other words in a given folder, and the second term, F^{coll}, relates word *w* with all other existing folders or groups. This way, if $f_j(w)$ corresponds to the number of times word *w* appears in folder *j*, i.e., the frequency of word *w* in *j*, then $F_j(w)$ represents the relative frequency of word *w*, defined as:

$$F_j(w) = \frac{f_j(w)}{\sum_v f_j(v)}; \quad v \neq w \quad \textbf{(2)}$$

It is important to note that $0 < F_j(w) < 1$ and $\Sigma_w F_j(w) = 1$. This normalization serves the purpose of dismissing the number of words in the folder and, instead, measure the relative importance of a word compared to the others contained in the folder. The relative frequency $F_j(w)$ will play the role of $F^{cluster}(w)$ in Eq. (1). To determine the representativity of word *w* in all folders, $F^{coll}(w)$, we use the following equation:

$$F^{coll}(w) = \frac{F_j(w)}{\sum_i F_i(w)}; \; i \neq j. \quad \textbf{(3)}$$

This way, it is possible to determine the *goodness G* of a word *w* that appears in folder *j* as:

$$G(w,j) = F_j(w) \frac{F_j(w)}{\sum_i F_i(w)}. \quad \textbf{(4)}$$

Words with a *goodness* value greater than a pre-specified threshold θ are allowed to enter the library of keywords. This is performed for all words of each document in all folders.

4. GENETIC ALGORITHMS AND WEB MINING

Genetic algorithms (GAs) are search and optimization techniques inspired by evolutionary biology [16,17]. Implementing a standard GA [18] starts with the creation of a random population of chromosomes, which are attribute strings or vectors with each attribute known as a gene. Individual chromosomes correspond to candidate solutions to the problem at hand, and are then evaluated and associated with a probability of selection and reproduction. Over the generations, individuals with high fitness values have higher probabilities of being selected for reproduction and thus propagating their genetic material throughout the population. In the standard GA, introduced in [16], a single population of individuals is available for evolution. This work proposes the use of a GA with two co-evolving populations to filter the information retrieved by the Google engine.

4.1. Related Works

Some approaches using Genetic Algorithms in web mining can be fond in the literature. In [19], the authors proposed a method for guiding genetic algorithms to perform information retrieval by fuzzy classification and genetic feature selection of terms from documents evaluated by the user. GeniMiner[20] is a genetic algorithm that manages a population of pages and aims at maximizing a fitness function that is mathematically based on the user

query. In [21] a personal agent that mines web information sources and retrieves documents according to user's interests was developed using classical information retrieval techniques and a genetic algorithm to learn and adapt to changes in the user's interests. The SmartSeek [22] employs a genetic algorithm to adapt to the user interest. The system accepts user feedback for fitness evaluation. In [23], the authors showed how to apply a genetic algorithm for mining student information obtained in a Web-based Educational Adaptive Hypermedia System. Agents based on genetic algorithms are presented in [24] to improve the performance of a self-organizing information retrieval network.

4.2. The GA Proposed

In the GA used here for web search and information filtering, there are two populations of chromosomes to be coevolved. In the first population each chromosome is composed of a pre-defined number of words randomly chosen from the keywords library of each folder. The chromosomes of the second population contain the same number of genes of the first population and the same number of individuals of the previous generation. Each gene of the second population may assume one of the three Boolean values (randomly chosen): AND, OR or NOT.

At each generation, the chromosomes of each population are concatenated in order to form a web query that will be used by the Google search engine to search for new documents in the web. The example shown in Fig. 1 illustrates the encoding scheme and a query generated by the GA using the chromosomes presented.

When no Boolean operator appears explicitly between two words in the query, there is an AND operator connecting them; the symbol "–" represents the NOT (negation) operator. The search agent uses this query to search for a document using Google. The document retrieved will be used to determine the fitness of this chromosome. To determine the fitness, the cosine measure [25], which determines the similarity between two vectors independently of their magnitude, was used. One vector represents the library of keywords in the folder, and the other represents the collection of keywords extracted from the document retrieved using the query. Eq. (5) returns the angle between these two vectors. It is equal to 1 when the vectors point in the same direction, and zero when they form a 90 degrees angle:

$$sim(D_D, D_Q) = \frac{\sum_{k=1}^{N} W_{Dk} W_{Qk}}{\sqrt{\sum_{k=1}^{N} W_{Dk}^{\ 2} \sum_{k=1}^{N} W_{Qk}^{\ 2}}}, \qquad \textbf{(5)}$$

where W_{Dk} is the frequency of word k in the keywords library of folder D, W_{Qk} is the relative frequency of word k in document Q.

After determining the fitness of all individuals, a binary tournament is performed to select those individuals that will compose the next generation [15],[16]. The best individual of the population is maintained and is not subjected to crossover. The crossover operator implemented here was the single-point crossover with probability *pc*. The only constraint of the crossover operator is that the same word cannot appear twice in the same chromosome. In such cases, the crossover operator is not applied and the parent chromosomes remain unchanged. In the current implementation no mutation is applied.

Query: results set OR genetic OR selection OR generation –evolution –individual –crossover filetype: PDF

results	set	genetic	selection	generation	evolution	individual	crossover
AND	OR	OR	OR	NOT	NOT	NOT	NOT

Figure 1 – Chromosome representation of the two populations used in the GA for web search. The top chromosome is composed of words taken from the library of keywords, and the bottom one is built by randomly choosing one of the three Boolean operators: AND, OR or NOT.

5. PERFORMANCE EVALUATION

In order to assess the performance of the genetic algorithm for information filtering (optimize web document retrieval), a benchmark database was used. The GA was tested in an environment containing three groups (communities): evolutionary computation group (EC), artificial neural networks group (ANN) and fuzzy systems group (FS). The PDF files used in the computer experiments were copied from the WCCI – IEEE World Congress on Computational Intelligence 2002: Proceedings of IJCNN 2002, FUZZ-IEEE 2002 and ICEC 2002. Table 2 summarizes the information about each of them.

Initially, the keyword extraction process (Section 3.1) was used to determine a number of keywords from each folder. The value used for the threshold was $\theta = 5\times10^{-5}$. This value was chosen empirically, and we could observe a trade-off between the value of θ and the number and quality of the keywords selected. High values of θ result in few words selected with high goodness values, whilst low values of θ result in many words selected with low goodness values. It can be observed, from Table 2, that for $\theta = 5\times10^{-5}$, 36 keywords were selected for the EC community, 37 for the ANN community and 92 for the FS community.

Table 2 – Information about the documents stored in each group.

Group	Number of papers	Total of words	Number of words in the group profile
EC	347	60,348	36
ANN	519	75,761	37
FS	285	37,876	92

For the tests performed, the GA used the following parameters (for both populations, words and Boolean operators): (i) number of generations: 20; (ii) population size: 20 chromosomes; (iii) chromosome length: 6 genes; and (iv) crossover probability: 60%.

Figure 2 shows the evolution of the fitness of the best individual (query) of the population and the average fitness of the population. It can be observed that the genetic algorithm is capable of improving by about 65% the quality (fitness) of the best individual, from the first to the last generation. It can also be seen that the diversity of the population at the end of the evolutionary process is quite low – this is indicated by the value of the average fitness of the population. This suggests that the use of a mutation operator may be helpful to insert and maintain the diversity of the population. Table 3 presents some examples of the types of documents retrieved by the queries evolved by the genetic algorithm.

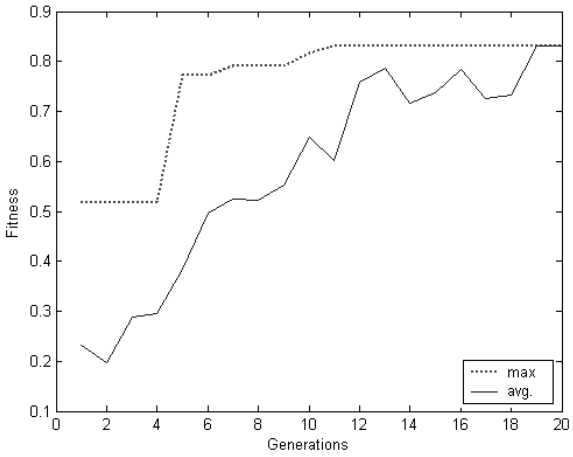

Figure 2 – Evolution of the best individual of the population (top curve) and the average fitness of the population (bottom curve).

Table 3 – Examples of documents retrieved by the search agents.

EC	Set of slides by S. Reid titled "Evolutionary Problem Solving". Retrieved from: http://nago.cs.colorado.edu/~strohman/evolutionary.pdf
ANN	Letter by S. Shtovba and Y. Mashnitskiy titled "The Backpropagation Multilayer Feedforward Neural Network Based Competition Task Solution". Retrieved from: www.liacs.nl/~putten/library/cc2000/SHTOVB~1.pdf
FS	Letter by S. Altug, H. J. Trussell and M.-Y. Chow titled "A 'Mutual Update' Training Algorithm for Fuzzy Adaptive Logic Control/Decision Network (FALCON)". Retrieved from: http://www4.ncsu.edu/~chow/Publication_folder/Journal_paper_folder/1999_NN_Mutual_update_Altug.pdf

6. CONCLUSIONS AND FUTURE TRENDS

This paper described a system for automatically generating group profiles for web documents by selecting a suitable library of keywords, and a search agent that generates and optimizes, via a genetic algorithm (GA), search queries for the Google search engine.

To illustrate the performance of the system it was applied to a data set containing three pre-defined user profiles: an artificial neural networks group, an evolutionary computation group, and a fuzzy systems group. These groups, or communities, were taken from the WCCI 2002 CD ROM for benchmarking purposes. The preliminary results presented demonstrated that the system is already capable of selecting appropriate libraries of keywords and, from them, making and optimizing queries for retrieving related documents from the web. It is important to remark that the usefulness of a retrieved document is not only related to the words that compose the evolved query, but also to all relevant words contained in the retrieved document.

There are still several avenues for future research. At first, we will implement Porter's algorithm [26] and include the mutation operator in the genetic algorithm in order to introduce and maintain the diversity of the population. Although it may look as if the current GA implementation does not have any type of diversity introduction mechanism, not always the search engine is capable of retrieving a document for a given query (the document may not exist or be available at a given time). In these cases, a new randomly generated query (chromosome) is created and introduced into the population, such that no individual with fitness value of zero is allowed into the population. This process of randomly generating new individuals does introduce diversity into the population of queries.

In order to build the whole academic virtual community, several other parts of the system must be implemented. For instance, a classifier agent will have to be designed so as to automatically classify the retrieved documents; and an interface agent, responsible for representing the interests of the user in the community (i.e., through this agent the user manifests its interests and preferences) will also have to be designed and incorporated into the system.

REFERENCES

[1] H. Lebo, *The UCLA Internet Report – Surveying the Digital Future,* UCLA Center for Communication Policy http://www.ccp.ucla.edu, 2003.

[2] B. Hermans, *Intelligent Software Agents on the Internet: An Inventory Offered Functionality of (near-)Future Developments.* Neverthelands, 1996. PhD Thesis, Computer Science Faculty, University of Tilburg, 1996.

[3] N. J. Belkin and W. Bruce Croft, *Information Filtering and Information Retrieval: Two Sides of the Same Coin?* Vol 35, Issue 12, December 1992 ACM Press New York, NY, USA

[4] B. D. Sheth, *A Learning Approach to Personalized Information Filtering*, M.Sc. Dissertation, Computer Science and Engineering, MIT, 1994.

[5] H. Rheingold, *The Virtual Community. Electronic version* http://www.rheingold.com/vc/book/ 02/02/2004.

[6] T. W. Yan and H. G. Molina, *The SIFT Information Dissemination System*, vol 24, Issue 4, December 1999 ACM Press New York, NY, USA

[7] G. Fischer and C. Stevens *Information access in complex, poorly structured information spaces* CHI'91, Human Factors in Computing Systems, 1991 pages 63-70 New Orleans, Louisiana, USA ACM Press.

[8] WEBKDD 1999 *Workshop on Web Usage Analysis and User Profiling* San Diego CA USA 1999.

[9] J. M. Barrie and D. E. Presti, *Colaborative Filtering* Science vol. 274, pp.371-372 1996.

[10] R. Cooley, B. Mobasher and J. Srivastava, *Web Mining: Information and Pattern Discovery on the World Web* Proceedings of the 9th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'97), November 1997

[11] S. Lawrence, K. Bollacker and C.L.Giles, *Indexing and Retrieval of Scientific Literature.* In Proc of the CIKM99, Kansas City, Missouri, (1999) November 2-6, pp. 139-146.

[12] S. Lawrence, C.L.Giles and K. Bollacker, *Digital Libraries and Autonomous Citation Indexing.* IEEE Computer, (1999) vol. 32, pp. 67-71.

[13] S. Lawrence, Online or invisible?. *Nature* (2001), vol 411, p. 521.

[14] A. Odlyzko, *The rapid evolution of scholarly communication.* http://www.citeseer.com 20/01/2004.

[15] K. Lagus and S. Kaski, *Keyword selection method for characterizing text documents maps.* Artificial Neural Networks, (1999), vol 7, pp. 371-376.

[16] T. Bäck, D. B. Fogel and Z. Michalewicz, *Evolutionary Computation 1: Basic Algorithms and Operators*, Institut of Physics Publishing 2000.

[17] T. Bäck, D. B. Fogel and Z. Michalewicz, *Evolutionary Computation 2: Advanced Algorithms and Operators*, Institute of Physics Publishing 2000.

[18] J. J. Holland, *Adaptation in Natural and Artificial Systems*, MIT Press 1975.

[19] M. J. M. Bautista, M. Amparo, V. Henrik, L. Larsen, *A genetic fuzzy classifier to adaptive user interest profiles with feature selection,* Proc. of the European Society for Fuzzy Logic and Technology (Eusflat-Estylf), Joint Conference, pp. 327-330, Palma, Spain 1999.

[20] F. Picarougne, N. Monmarché and A. Olivier, G.Venturini, *Web mining with a genetic algorithm* WWW2002 The Eleventh International World Wide Web Conference, Honolulu Hawaii USA May 2002.

[20] M. S. Valim and J. M. A. Coello *An Agent for Web Information Dissemination Based on a Genetic Algorithm* In: IEEE International Conference on Systems, Man and Cybernetics- IEEE SMC'03, 2003, Washington, D.C., USA Proc. IEEE International Conference on Systems, Man and Cybernetics- IEEE SMC'03. IEEE Press, 2003. p.3834 – 3839

[21] A. Joshi and S. Todwal *Evolutionary Machine Learning for Web Mining* TENCON 2003 Bangalore - India October 2003

[22] C. Romero, S. Ventura, C. Castro, W. Hall and M.H. Ng, *Using Genetic Algorithms for Data Mining in Web-based Educational Hypermedia Systems*
http://w ww.lcc.uma.es/~eva/WASWBE/romero.pdf

[23] K. Abe, T. Taketa and H. Nunokawa, *An Efficient Information Retrieval Method in WWW Using Genetic Algorithms* ICPP Workshops 1999 Fukushima Japan.

[24] Z. Z. Nick and P. Themis, *Web Search Using a Genetic Algorithm.* IEEE Internet Computing (2001) vol 5, pp. 18-26.

[25] G. Salton and M. J. McGill, *The SMART and SIRE Experimental Retrieval Systems* in Readings in Information Retrieval. Morgan Kaufmann Publishers Inc. (1997), pp. 381-399.

[26] M. Porter, *An Algorithm for Suffix Stripping,* Program. (1980), vol. 14, no. 3, pp. 130-138.