An Adaptive Multi-Agent Based Search Engine

H.O.Nyongesa S. Maleki-Dizaji, Z. Aliothman
School of Computing and Management Science
Sheffield Hallam University, S1 1WB UK

Abstract
The current exponential growth of Internet information content demands improved tools to help cope with the volume of information available. As the volume and variety of available information continues to grow, it is increasingly difficult to obtain information that accurately matches user needs. This is due, firstly, to the fact that users often do not present to information retrieval systems queries that optimally represent the information they want, and secondly, the measure of a document's relevance is highly subjective and variable between different users. This paper addresses this problem with an approach that relies on evolutionary user-modelling, in order to retrieve domain-specific information. The paper describes an adaptive information retrieval system that learns user needs from user-provided relevance feedback. The method combines qualitative feedback measures using fuzzy inference, and quantitative feedback using genetic algorithms (GA) fitness measures. An autonomous agents design approach is applied.

Keywords: Information Retrieval, Genetic Algorithms, Fuzzy Inference, Relevance Feedback, Multi-Agent System

1. INTRODUCTION
An optimal Information Retrieval System (IRS) is one which is able to retrieve from a database, only those documents that are relevant to a user's information needs, but excluding documents that are irrelevant. The concept of relevance, however, is one that is subjective and influenced by a variety of factors. First and foremost, the queries posed to information retrieval systems are in most cases not optimal in terms of describing the required information, with respect to an individual user's information needs. Among the factors that influence the relevance of a retrieved document are: user knowledge level, user perception, information currency and clarity. Consequently, in spite of great improvements in the efficiency of search engines many return a large proportion of non-relevant documents. Thus recently, there has been a definite paradigm shift from a view of relevance as simple term-matching between query and document, to a view of relevance as a cognitive and dynamic process involving interaction between the information user and the information source. The most common performance metrics for IR systems are based on recall and precision rates. Recall measures the proportion of relevant documents in a document corpus that is retrieved, while precision measures the proportion of retrieved documents that is relevant. However, both performance criteria are particularly difficult to measure due to the large variability in the relationship between information content and user-needs context. This paper proposes that these issues can be dealt with by adding some "intelligence" to an IRS, based on ability of the IRS or information source to build a representation of user information needs.

There are two conventional approaches to developing user models in IRS: (1) a knowledge-based approach [1] and a machine-learning approach [2]. The knowledge-based approach involves endowing the IRS with a great deal of domain-specific background knowledge about different users, gleaned from human experts, to be able to create clusters or stereotypes of users. Such knowledge is used to customise the dialogue with the user. The main disadvantages of this approach are that the system is only able to perform what it is programmed to do, and significant effort and a priori knowledge is required from the domain experts. In the machine learning approach, the IRS is designed to autonomously acquire knowledge through interactions with the user. The advantages of this approach are that it requires less effort from users and the system is able to adapt to changes in user-needs over time. The use of machine learning techniques for generating and adapting user models in information retrieval applications is therefore compelling. So far, however, the performance of such approaches is still low.

Several techniques have been proposed for this purpose, including relevance feedback (RF) [3], symbolic learning [2], Genetic Algorithms (GA) [4], simulated annealing [5] and fuzzy logic [5]. We present a novel machine learning approach, which not only learns to retrieve relevant information, but also continuously adapts after every search to improve the quality of future search results. It aims to increase retrieval precision, with minimum penalty on recall, by adapting to new information needs within a specific subject area over a period of time. This is achieved by applying a fuzzy relevance feedback and evolutionary learning with GA. Furthermore, the IRS has been developed on a multi-agent paradigm representing the different typical activities, including document representation, query formulation, user-needs modelling, and user-needs model reinforcement. The use of a multi-agent approach also offers a benefit of parallelism, in order to reduce execution time. The system thus, is made up of several agents, each one specialising in a particular activity and includes: (1) interface
agent, (2) search agent, (3) filter agent, (4) user-model agent and (5) document agent. The schematic diagram is shown in Figure 1.

2. OVERALL SYSTEM FUNCTIONALITY AND DESIGN

The proposed IRS is a personalisable adaptive tool for information retrieval from the Internet. In this regard, the "black-box" view of the system consists of a document collection from the Internet as input, while the output are processed documents ranked according to user interests and preferences. In order to achieve the system goal, the user is required to communicate with the system, via an interface agent with two objectives: (1) provide interest keywords and (2) provide relevance feedback of the retrieved documents. The search agent is a meta-search engine of any Internet sources, which uses interest keywords to retrieve documents. The document agent normalises and indexes the retrieved documents using stemming algorithms [6]. In order to increase the precision of retrieved documents, a filter agent ranks the indexed document according to an existing user-needs model. The ranked documents are then passed back to the user through the interface agent, which allows the user to evaluate the relevance of the documents, by giving a qualitative score to each document. In order to effect a perpetually evolving user-needs model, the user model agent maintains a "population" of competing interest keywords, which are used by the filter agent to rank the retrieved documents. These are also the chromosomes, which evolve using GA. The overall agent interaction diagram is shown in Figure 2. JADE was considered to be the most suitable tool for creating the multi-agent IRS. JADE is integrated with JESS, a Java shell of CLIPS, in order to exploit its reasoning capabilities.

3. MODEL AND DOCUMENT REPRESENTATION AND SIMILARITY

The most commonly used method for representation of documents in IRS is a set of features derived from the document collection, typically, a list of indexed keywords. However, there exists a large class of words that have no inherent meaning when taken out of context and, thus, are basically useless as index terms. For example, terms such as 'the', 'and', 'or', 'of' have no semantic content. These words also tend to be among the most frequently occurring terms in the English language. Therefore, it is reasonable to filter such terms out of index terms. This is done by creating a list of terms that do not need to be indexed, otherwise known as a stop list. Use of a stop list can significantly improve the efficiency of an IRS by reducing both the size of index terms, and the time required to conduct a search. Another strategy for reducing index size is to create word equivalence classes for index terms by removing and modifying prefixes and suffixes to identify the root form of the word, using stemming algorithms [6]. A further refinement in the representation of documents is the assignment of numerical weights to index terms in a document representation. The weight assigned to a term occurring in a given document is an attempt to quantify that term’s importance to the subject matter of that document. There are many methods for calculating the weights, the most common ones being statistical based on the terms' frequency of occurrence within a document, known as the term frequency (tf), or within a corpus of documents, known as the document frequency (df).

In this paper, we have defined an information-theoretic entropy function for weight assignment: Thus, the weight \( W_{i,d} \) of a term \( i \) in document \( d \) is given by,
where \( tf_i \) is the number of times word \( t_i \) appears in document \( d \) (the term frequency), \( df_i \) is the number of documents in corpus which contain \( t_i \) (the document frequency), \( n \) is the number of documents in the corpus and \( tf_{\text{max}} \) is the maximum term frequency over all words in \( d \). A document representation therefore, comprises a finite list of keywords, each associated a weight.

Improving the precision of retrieved documents is possible dependent on an ability to accurately measure the similarity between a query vector (user-needs) and a document vector (retrieved documents). This is, moreover, complicated by the fact that the measure of the relevance of a document to a user is not objective, and therefore difficult to express quantitatively. We have proposed to overcome this by applying two methods to determine the similarity; a quantitative (objective) measure based on term weights, and a qualitative (subjective) measure based on user relevance feedback. The qualitative measure is described in the next section. A quantitative measure is determined by the inner product of the corresponding weight vectors. Thus, if query vector and document vector terms are assumed to be independent, the similarity between the vectors is given by,

\[
W'_{ij} = \frac{1}{2}(1 + \frac{tf_{ij}}{tf_{\text{max}}}) \frac{df_i}{n} (\log \frac{df_i}{n})
\]  

where \( tf_i \) is the number of times word \( t \) appears in document \( d \) (the term frequency), \( df_i \) is the number of documents in corpus which contain \( t \) (the document frequency), \( n \) is the number of documents in the corpus and \( tf_{\text{max}} \) is the maximum term frequency over all words in \( d \). A document representation therefore, comprises a finite list of keywords, each associated a weight.

Improving the precision of retrieved documents is possible dependent on an ability to accurately measure the similarity between a query vector (user-needs) and a document vector (retrieved documents). This is, moreover, complicated by the fact that the measure of the relevance of a document to a user is not objective, and therefore difficult to express quantitatively. We have proposed to overcome this by applying two methods to determine the similarity; a quantitative (objective) measure based on term weights, and a qualitative (subjective) measure based on user relevance feedback. The qualitative measure is described in the next section. A quantitative measure is determined by the inner product of the corresponding weight vectors. Thus, if query vector and document vector terms are assumed to be independent, the similarity between the vectors is given by,

\[
Sim(a,b) = \sum_{i=1}^{n} \sum_{j=1}^{m} w_{ai} w_{bj} \forall (t_j = t_i)
\]

where \( w_{ai} \) are the weights of terms in the query vector (a) and \( w_{bj} \) are weights of similar terms in document vector (b).

\subsection*{4. EVOLUTIONARY REINFORCEMENT OF USER MODELS}

A proven method for automatically generating improved queries is the well-known relevance feedback [7]. Relevance feedback is a process of redirecting an IR system's output (documents) back into that system's input (query) to produce another, more accurate output (more relevant documents). The main assumption behind relevance feedback is that documents relevant to a particular query resemble each other in the sense that they are represented by reasonably similar descriptors [8]. A user indicates to the system which of the documents from the collection just retrieved are considered to be relevant to the query. The query formulation can then be improved by increasing its similarity to such previously retrieved relevant documents. One way of applying this method is to select a set of terms from the relevant documents and add to user queries, or by removing terms appearing in the non-relevant documents.

In this paper, a GA [9] is used to evolve and adapt query vectors, which are the models of user information needs. With GA, user models represent hypothetical knowledge about the user needs, encoded in a chromosome. These chromosomes are expressed in terms and their weights, unlike in previous studies [3] where only terms were used. Each chromosome, thus, is a hypothesis on how to evaluate the relevance of a document, and competes against other chromosomes to predict user satisfaction from the retrieved documents. It is assumed that the user information needs are stochastic, but non-transient. In other words, the information needs vary in a non-deterministic manner between users, but, they do not change rapidly over time. Or, it can be said that while the perceived relevance of the same documents may vary widely between different users, each user's perception of relevance of the documents does not change rapidly over time. In this regard, the GA can only be used to evolve a user model for one user (or specific group of user). The GA through user feedback, however, can effect an adaptation of user needs to new areas, thus, improving and maintaining the retrieval precision in real-time.

GA are blind search mechanisms, hence for online learning it is a advantageous to initialise the population with any a priori knowledge in order to avoid undesirable performance in the early generations. In this research, the initial chromosomes are comprise of terms from a vocabulary that was judged relevant for individual users, and the assignment of fitness for the initial population is proportional to the mean similarity between all chromosomes in the population set given by,

\[
f_0(q_i) = \frac{1}{n} \sum_{j=1}^{n} \text{sim}(q_i, q_j)
\]

modify the representation of user needs, based on a quantitative and a qualitative relevance metric. The qualitative metric is proportional to the mean similarity between the query vector and all (or ranked) retrieved documents. The qualitative metric relies on a fuzzy relevance feedback from the user. Thus, the user provides a linguistic value of feedback for a retrieval, which is used in a fuzzy inference system to obtain a crisp feedback measure. The crisp feedback is combined with the measure of similarity between the query vector and the retrieved document to determine the a quantitative metric used adjust the fitness of the chromosomes in the population.

Fuzzy logic [10] has been applied in this research because it provides a very convenient methodology for "computing with words" (linguistic as opposed to numeric values). Although words appear less precise than numbers, their use in information processing is closer to human perception of concepts. A fuzzy inference system (FIS) is a rule-based system that associates a set of inputs (conditions) and a set of outputs (actions), as shown in Figure 3 with reference to the current application. The rules for the FIS are shown in Table 1. Each of the cells in the Table represents an IF
<conditions> THEN <action> statement. For example, the first cell in Table 1 is a statement: IF<Similarity = P; Feedback = E> THEN <Adjustment = IH>, where P, E, IH stand for the fuzzy linguistic values "Poor", "Excellent" and "Increase High", respectively. These rules, in general, are heuristic and rely on obtaining knowledge from a domain expert. In the fuzzy system, the universes of discourse over which these linguistic values are defined are, similarity ∈ (0.0 ... 1.0) and feedback ∈ (-3...3) and adjustment ∈ (-0.4 .... 0.4). An example of the specification of the linguistic values, for similarity, is shown in Figure 4. Detail of the operation of FIS is provided by Lee [12].

The output of the inference system is used adjust the fitness of the chromosomes competing to evolve the optimal user-needs models. However, since only one chromosome is, in fact, used to retrieve documents (evaluated), the fitness of the rest of the population is adjusted in proportion to their distance from the evaluated chromosome. This is given by,

\[ f(q_j) = (1 - \beta) f(q_0) + \alpha \sum_{i=1}^{n} \left( \text{Sim}(Q_0, d_i) \right) f(q_i) \text{Sim}(Q_0, q_j) \quad j = 1 \cdots n \]  

where \( f(q_j) \) is the old fitness, \( \beta \) is a fitness sensitivity factor that limits the change and maximum value of fitness, \( Q_0 \) is the evaluated (query) chromosome, \( q_j \) are the rest of the chromosomes in the population, \( d_i \) are the retrieved documents and \( \alpha \) is determined from fuzzy inference system. The output of the FIS is also used to adjust the term weights in the chromosomes. This is due to the observation that the initial term weights, as calculated using term frequencies, may not in fact be an accurate representation of importance of the documents to the user. Hence, the need to update with actual user relevance feedback. Thus, weight adjustments are given by,

\[ w'(q_j) = w(q_j) + \alpha \omega(d_0) \quad \forall t_j = t_0 \]  

where, \( d_0 \) is the retrieved document vector, \( q_j \) are the query chromosomes and \( t_j \) the terms in the query and document vectors. The resulting effect is that, for those terms already present in user model, term weights will be modified by the feedback and terms not already present in the model may be added to it. Terms are also removed from the model representation when their weights are zero or less, and the total number of terms is limited by the predefined chromosome size.

5. EVALUATION AND RESULTS

The performance of system was measured based on recall and precision values. Precision measures the quality or efficiency of the search, in terms of the fraction of useful material in the set of retrieved documents. Recall measures the breadth of the search, expressed as the fraction of the available relevant material presented to the user. In addition to retrieval performance, the performance of the machine learning techniques (GA and fuzzy logic) was measured over time to tune the learning parameters.

After setting the parameters, the precision and recall were measured to evaluate the performance of the IRS. The IRS was tested for user-needs in four different subject domains, namely, Information Filtering, Agents Technology, Fuzzy Logic, and Genetic Algorithms. Each query was designed to retrieve 10 "best documents", which were ranked by an experience user in each of the subject areas for 10 successive retrievals. The precision rates were measured after retrieval. Results are shown in Figure 5.

It is seen that there was improvement in the precision rates for all four subject domains. However, two subject areas seemed to experience larger precision increases than the other two. On examination, it is clear that these subject areas, in fact, are quite specific, and the retrieved documents were successively ranked as relevant. The area of Genetic Algorithms, for example, while being a specific discipline does attract retrieval of documents from biological disciplines.

7. DISCUSSION AND CONCLUSION

In this paper we have described a novel approach for improving document retrieval efficiency by combining fuzzy relevance feedback and evolutionary reinforcement. The approach has been designed and tested, especially for applications where user information needs are subjective but relatively static, and hence can be accurately modelled. This includes special-interest information retrieval, such as, academic research documents. The representation and adaptation of user models is achieved through the processes of Genetic Algorithms. Special GA operators suitable for information retrieval applications have been investigated.
Initial results, on test documents, has shown that significant improvement can be achieved over conventional query and retrieve techniques, or the use of relevance feedback alone. In particular, the precision of information retrieval is shown to increase significantly over conventional search tools. While Vrajitoru (1998) found that the GA was less efficient than relevance feedback in a similar application, our result indicates that the combination of both is in fact advantageous. Our continuing research, it is envisaged will in due course validate this claim.

**Figure 3: Fuzzy Inference System**

**Figure 4: specification of linguistic values**

Similarity Values: Poor (P), Moderate (M), Good (G), Very Good (VG) Feedback Values: Excellent (E), Very Good (VG), Good (G), Moderate (M), Poor (P), Very Poor Output Values: Increase High (IH), Increase Moderate (IM), Increase Low (IL), Decrease Low (DL), etc.

**Table 1: FAM Table for User Model Adjustment**

<table>
<thead>
<tr>
<th>Feedback</th>
<th>E</th>
<th>VG</th>
<th>G</th>
<th>M</th>
<th>P</th>
<th>VP</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>IH</td>
<td>IH</td>
<td>IM</td>
<td>HI</td>
<td>DL</td>
<td>DM</td>
</tr>
<tr>
<td>M</td>
<td>IH</td>
<td>IH</td>
<td>IM</td>
<td>HI</td>
<td>DL</td>
<td>DM</td>
</tr>
<tr>
<td>G</td>
<td>IM</td>
<td>HI</td>
<td>IL</td>
<td>DL</td>
<td>DL</td>
<td>DH</td>
</tr>
<tr>
<td>VG</td>
<td>HI</td>
<td>HI</td>
<td>DL</td>
<td>DM</td>
<td>DH</td>
<td>DH</td>
</tr>
</tbody>
</table>

**REFERENCES**