Behavioral Model Extraction of Search Engines Used in an Intelligent Meta Search Engine

KAVEH KAVOUSI
Computer Department, Azad University, Garmsar Branch
BEHZAD MOSHIRI
Electrical and Computer department, Faculty of engineering, University of Tehran
IRAN

Abstract: Information fusion placed over the Data fusion level prepares grounds to gain more perfect and clear results based on uncertain collected information on one subject from different aspects. Nowadays, the need for intelligent systems as personal Meta Search engine capable of supplying user by needed information from great mass of information resources is sensible. More over, the measures taken on this ground have many deficiencies. In a Meta Search engine the user's interests are received and the proper queries based upon them are transmitted to the search engines. Then, the returned results of the search engines become filtered and based on priority they are made available for the user. But, it is obvious that the different search engines have different behavior on different subjects. On the same direction in this study we try to examine a part of a customized intelligent agent which is able to extract behavioral model of search engines from different subjective clusters gradually, and according to the feedback it gets from the user.

Key words: Data/Information fusion-list fusion-Meta search engine- Intelligent agent – Subjective clusters.

1 Introduction

Undoubtedly,suitable retrieve of information from internet and other data sources with large scales and very large scales is one of the most important problems in efficient use of information sources. Nowadays, web is the largest data source of documents and other forms of information, and a suitable ground for evaluating the different Information retrieval techniques. The more the web is expanded, the more need for powerful search tools become evident. At the present, there are lots of services for web search. But none of them are helpful as expected, and actually in the most cases the results are dissatisfactory. One of the most important reasons of this is because of inaccurate knowledge of the users about present search engine abilities, by the other word, their behavioral model. Researches done about the internet Meta Search engines shows that the **lists fusion** which has its own independent literature [11] and also the behavioral model of internet search engines were not studied from the angle considered in this paper.

2 The Characteristics of Intelligent Meta Search engine based on Information Fusion

In this paper, an intelligent Meta Search engine is discussed which using information Fusion techniques, it roles as a customized Meta Search engine for the user. This agent receives the words or phrases interested for the user who is willing to find the subjects related to them to study, and then it asks the user to determine weights of words according to their importance, whether or not to be used in the text. Weight has a linguistic concept here. It means that the user can determine the importance of whether or not word to be in the text as **None, Low, Medium, High, and Very High**. Then the agent makes ready the queries, by a unit named "Query Generator", according to the number of information servers (e.g. internet search engines, and/ or data sources) [1,2].

After sending queries each server returns a list of ranked documents based on their proximity to the subject, and according to the algorithm that server works upon it.

Then the Meta Search engine reviews these lists and then eliminates the repeated items, and fuses them based on list fusion algorithms such that a ranked list of documents is prepared. In this list a

score is given to each document based on its position in the list [11].
Then the documents are processed one by one in this list, and their conditions are determined regarding whether or not the key words or phrases selected by the user to be present, and according to their presence quantity and distribution, two scores are given to each document. In this mechanism, Ordered Weighted Averaging operator (OWA) [4,8,9,10,12] has the basic role. The mentioned aspects of Meta Search engine operation are discussed in [1,2,3] thoroughly. This paper initially focused on the fusing method of the lists attained from search engines and modeling of the search engines used by Meta Search engine.
Each time the user decides to use the Meta Search engine, he or she specifies that this interesting subject is in which subjective cluster [6,7]. By **subjective cluster**, we mean "a logical classification of interesting subjects". Each time the user starts a new search; he or she can select from the available clusters or create a new cluster. Some of available clusters at the present are shown in Table 1. The agent has allocated one score to each server based on its historic operation in each subjective cluster. The agent allocates the score of the server which has retrieved the document to it.
These scores become updated after each use of Meta Search engine, based on an algorithm-thoroughly discussed later – so gradually the behavioral model of each search engine and its efficiency on a special subject is formed in the mind of intelligent Meta Search engine. Regarding the mentioned explanations, 4 different scores are gained for each document that at the end the agent must calculate the final score of each document based on them and represent it to the user. It is done by fusing of these scores by methods of information fusion and for each score a weight is considered which shows the importance of that criterion at the final decision. Also, finally, the score of each server in the special subjective cluster is updated with / without feedback received from the user.

3. The problem of list fusion

Different data sources on the web often complete each other. Thus, to cover all the information resources and to gain more pure results, it is a logic strategy to use different search tools and at the end the results to be refined and then fused together. Now, the question is: what is the best method to fuse these lists together? This question is important because the lists which are represented by the search tools are often ranked. Now, we want to fuse all these lists together to get a unit list which its items are selected from all the presented items in all the lists. But fusing these and producing a final list is an important discussion.
Suppose we have a group of information servers. We show these servers with $\begin{cases} S_i \\ i = 1,...M \end{cases}$ In which, M is the number of servers. Also, we suppose each server got a unique collection of documents. It means that each document is in just one information server. Of course, we can't suppose this for internet search engines. But we can create the same condition by eliminating the repeated items from other lists. We also suppose that each information server has its own search mechanisms. For a Query Q, each server give one score to each document and at the end prepares a ranked list of documents related to the query as its answer.
The problem we have to solve: to choose N documents most related to query and put them into the final list. Each N document may be got from each of the servers. The point which makes our job harder is the difference among servers in their methods of allocating scores, and these methods may never be comparable. For this reason we can't select N documents in order from highest score to the lowest one in each server.

3.1 Representing Mathematical Formulation of the Lists Fusion Mechanism

Let us denote the number of lists that we want to fuse by M.The lists themselves can be denoted by $L_1,\ldots,L_M$. For each j from 1 to M let us denote by N_j the number of items in j-th list. A natural way of fusing these M lists into a single sorted list is to assign a value v to each item of the lists and then sort all $N_1 + N_2 + \ldots + N_M$ elements in the increasing order of this value. So the question is: How can we determine the value v for each item? We need a function of two variables. An item is uniquely characterized with two parameters:

- The number j of the list L_j from which this item comes
- The order i of the given item in the corresponding list L_j $(i = 1,2,\ldots)$.

The value v must be uniquely determined by the values of these two parameters. Dependence v on j, can only appear through dependence on N_j.So v is a function of $i$and N_j .

(1) $v = v(i, N_j)$

If two lists L_k and L_j have the same length, there is no reason to assign higher priority to each one of them. It means: $v(i, N_j) = v(i, N_k)$. The way of calculating $V(i, N_j)$ is described thoroughly in [1]. Here, we again state the optimal formula for such function is:

2) $v(i, N) = N^{\alpha}.(i + cN)$

In the same reference, it is shown that for $\alpha = 0$ also this formula can model the behavior of an expert person. Now, after utilizing fusing lists method, we have a ranked list, that each document in it has a score based on its rank in this list.

A simple and adequate method is represented later. But, in advance we represent the following definitions:

- Total number of documents = N
- Absolute score of document k in the fused list = μ_K . This score is described completely before.
- Normalized score of document k = V_K

This is one of the four criteria in final documents scoring and for each document is calculated as:

$$V = \frac{\mu_K}{Max_{i=1}(\mu_i)}$$

4. Score allocation to information servers

As explained before now we have a ranked list of documents which extracted from different search engines. But, in this ranking the behavioral model of search engines has no role. But as it is explained before due to variability and differences of design parameters and the designers of each search engine, and also due to aims of each search engine, each search engine has a powerful function in one Field, and medium or weak in another. Thus, neglecting this fact leads to accuracy decreasing in Meta Search engine's function. In this part, the place and time are given to extract behavioral model of each search engine to utilize it in documents ranking.

Initially, a score of 0.5 is given to all the information servers. This includes the sources will be added gradually to the system in the future. It must be considered that the ideal score of a source is 1. Each time the user sends a request to the system and gets result (s) from it, a list of ranked documents is prepared. We devised a method for learning the **importance weight** of each information server. This parameter has important role in making behavioral model of the information servers. A score is allocated to each document in the final list based on the information server that has retrieved that document. Thus, the documents retrieved from servers with more powerful background in that certain subject, have more chances. This point is important. Because, some information servers may be very powerful in a certain field (subjective cluster), or they are designed and practiced for retrieving the documents related to a certain subject.

For modeling of score allocation to each information server we represent the following definitions:

- The number of information servers which have at least one document in the final list = M
- The number of documents presented from server i in the final list = d_i
- Set of the documents ranks related to server i arranged in increasing order = R_i

$$R_i = \{ r_k \mid k = 1, \ldots, d_i ; r_1 < r_2 < \ldots < r_{d_i} \}$$

- The number of the documents with most importance in the final list = K

(The user will check only K documents in the final list)

- The score of Server i form beginning till now for cluster j = $s_{ij}^{(t)}$ $(0 \le s_{ij}^{(t)} \le 1)$
- The score of Server i in the next step for cluster j = $s_{ij}^{(t+1)}$ $(0 \le s_{ij}^{(t+1)} \le 1)$
- The absolute score of Server i in the current step for cluster j = ϕ_{ij}
- The relative score of Server i in the current step for cluster j (resulted from normalizing φ_{ij}) = ψ_{ij} $(0 \le \psi_{ij} \le 1)$

Now, we explain the calculating way of ϕ_{ij} . It is observed that at the present time each information server's score to the subjective cluster is $S_{ij}^{(t)}$ which is between 0 and 1, and this value is considered for all the documents retrieved by this server, and along with other scores participates in the final score(using OWA operator). To calculate ϕ_{ij} variety of methods can be adopted. But, we should find a moderate method to this figure. It seems that the following methods are adequate:

$$(5)\ \phi_{ij} = \sum_{r_k \in R_i} r_k^{-\beta(K)} = \sum_{\substack{k=1 \\ r_k \in R_i}}^{d_i} r_k^{-\beta(K)}$$

(6) $\phi_{ij} = \sum_{r_k \in R_i} e^{-\frac{r_k}{\beta(K)}} = \sum_{\substack{k=1 \\ r_k \in R_i}}^{d_i} e^{-\frac{r_k}{\beta(K)}}$

In each of the above formulas the more the document become far from the top of list the more the allocated scores decreased, and the final score of each server is obtained from sum of scores of documents that are retrieved by this server.

$\beta(K)$ is a continuous function of K in which K is the number of document most considered by the user in the final list. For example, $K = 10$ means that ordinarily the first 10 documents is more useful for the user, thus information servers from which the first 10 documents are retrieved must get the most increasing in scores. The value of K is determined by the user. $\beta(K)$ specifies the documents, How affect on score changing of their information servers. For example in (5) the bigger $\beta(K)$ conduce that documents with higher rank be more effective in increasing the score of related information server. Relating to (5) if $\beta(K) > 1$, the results will be unreasonable, such that there will be a big difference between document with first position in the list and the second. Relating to (6) this is vice versa. The more $\beta(K) > 1$ is the more the condition is moderate. To make the matter more clear an example represented:

Example: suppose we have 5 information servers (M=5) which are specified by S_1 to S_5. Also, suppose that the fused list for subjective cluster j is as Table 2. Considering table 2 the values of d_i and r_i are calculated according to table 3 ($S_{ij}^{(t)}$ is the score of Server i form beginning till now for cluster j). Now, relating to (5), for $\beta(K) = \frac{K}{20}$ and $\beta(K) = \frac{K}{5}$ and K= 10 we calculate ϕ_{ij} (Table 4).

As it is observed, the values of ϕ_{ij} for $\beta > 1$ are exactly on the opposite of an expert's view. Because, considering the ranked list of documents, an expert evaluates the scores of S_2 and S_5 servers close to each other, but for (5) when $\beta > 1$ this is not correct. Also, for $0 \le \beta \le 1$, determining the suitable amount for β is not simple. We can show that (5) is not suitable for our purpose. But tuning τ in (6) can produce proper results (Because of the nature of function $\begin{cases} f(t) = e^{-\frac{t}{\tau}} \\ t \ge 0 \end{cases}$). Thus, it makes possible that only the first K documents in the final list increase the score of their related information server. For example, if an information server has retrieved even just the last document interested for user (interesting document K), it gets a positive score. But, after that the speed of decreasing the allocated scores will increase rapidly.

$\begin{cases} \beta(K) = \frac{K}{\alpha} \\ 3 \le \alpha \le 5 \end{cases}$ can be the simplest form for this purpose. Considering the above points the formula used by the intelligent agent to calculating the values of ϕ_{ij} is as following:

(7) $\phi_{ij} = \sum_{\substack{k=1 \\ r_k \in R_i}}^{d_i} e^{-\frac{r_k K}{\alpha}}$ ($\alpha = 3$ is suitable amount). Like other parts, with normalizing the amounts of ϕ_{ij}, we calculate the score of each server as following: (8) $\psi_{ij} = \frac{\phi_{ij}}{Max_{k=1}^{M}(\phi_{kj})}$

Now we describe the way of updating server's score by the agent. $S_i^{(t)}$ is the server i score on the current step. We are going to find a function by which we calculate $S_i^{(t+1)}$ (server i score on the next step): (9) $s_i^{(t+1)} = f(s_i^{(t)}, \psi_i)$

The score of each server can be updated by determining function f. But, it must be considered that the time parameter, also, affects the function, indirectly. The importance of this function is capability of it in reconstructing the behavioral model of each server. In this case, the simplest way, is calculating the average scores of each sever in each cluster. To do this the following formula is suitable: (10) $s_{ij}^{(t+1)} = s_{ij}^{(t)} . \frac{t}{t+1} + \frac{\psi_{ij}}{t+1}$

In which, t is the number of queries that agent has sent them about cluster j to the server i.

The results obtained from the above mechanism, improve the quality of the Meta Search engine's results effectively. The complete results of using this mechanism are represented in [1].

5. Conclusion

Since each of the internet search engines are produced by their own designer's thought, vision,

and reasoning it is obvious that they have different behaviors on fulfilling the users' demands on searching different subjects. Thus, it seems that designing an intelligent Meta search engine without considering behavior of each search engine against different subjects is inaccurate.

The results obtained from this paper shows that considering this parameter in designing Meta search engines, conduce to improvement in quality of output results of designed Meta Search Engine.

Accurate modeling of a search engine, in addition to stated parameters in this paper, may depend on other parameters, too. For example, many search engines consider the amount of money received form the document owner in final ranking, on which we didn't discuss. This parameter and the others effecting on modeling process can be studied more complete in next studies.

References:

[1] Improving the intelligent methods of Information Fusion Software agent on internet, Kaveh Kavousi, M.Sc thesis for Artificial Intelligence and Robotics, Dept. of Electrical and computer engineering , Faculty of engineering,University of Tehran.

[2] Improving the function of intelligent agent of information fusing, Kaveh Kavousi, Behzad Moshiri, Technology College of Tehran University's publishing, summer 2004 issue.

[3] Architectural designing of an intelligent agent based on data fusion for extracting information from searching fields, Behzad Moshiri, Kaveh Kavousi, The 11th electricity engineering Conference in Shiraz.

[4] A Broad Class of Standard DFSes, I. Glockner, Bielefeld University Report TR-2000, 2000.

[5] Using An Intelligent Agent to Enhance Search Engine Performance, J. Jansen, Peer–Reviewed Journal on the Internet, http://www.eecs.usma.edu/usma/academic/eecs/instruct/jansen/, 1997.

[6] Inductive learning from considerably erroneous examples with a specificity based stopping rule, J. Kacprzyk, Proceedings of the International Conference on Fuzzy Logic and Neural Networks, Iizuka, Japan, 819, 1990.

[7] Text-Learning and Related Intelligent Agents: A Survey, D. Mladenic, IEEE Intelligent Systems Journal, pp. 44-54, July/August , 1999.

[8] Piecewise Linear Aggregation Functions, S. Ovchinnikov, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, Vol 1. No.1 (2000). pp.11-22, 2000.

[9] Decision making under Dempster-Shafer uncertainties, Ronald R. Yager, International Journal of General Systems, Vol. 20, pp. 233-245, 1992.

[10] Fuzzy logic controllers with flexible structures, Ronald R. Yager and D. P. Filev, Proceedings of Second International Conference on Fuzzy Sets and Neural Networks, Iizuka, Japan, pp. 317-320, 1992.

[11] On the Fusion of Documents from Multiple Collection Information Retrieval Systems, R. R. Yager, A. Rybalov, Journal of the American Society for Information Science, 1997.

[12] Fuzzy quotient operators for fuzzy relational data bases, Ronald R. Yager, Proc. Int. Fuzzy Engineering Symposium, pp. 281-296, Yokohama, Japan, 1991.

Table 1: Information clusters list which the agent holds their background to identify behavioral model of information servers.

Name of cluster	Cluster No.	Name of cluster	Cluster No.
Mobile Robot Navigation	8	Data / Information Fusion	1
In tumescent Coatings	9	Context Sensitive Web Searching	2
Paint & Resin Technology	10	Dempster Shafer Theory	3
Robotics	11	Computer Science , Hardware	4
TBM model	12	Computer Science , Software	5
Neural Networks	13	Case Based Reasoning	6
Neuro Fuzzy Systems	14	Fuzzy Controllers	7

Table 2- fused list gained from ranked lists that retrieved by Information servers

the server that retrieved this dacument	Score	The server that retrieved this	Score	The server that retrieved this	document
S_5	11	S_1	6	S_2	1
S_4	12	S_3	7	S_5	2
S_5	13	S_1	8	S_5	3
S_4	14	S_2	9	S_3	4
S_3	15	S_2	10	S_2	5

Table 3- Extracting the rank of each document in fused list

$s_{ij}^{(t)}$	$R_i = \left\{ r_k \mid k = 1, \ldots, d_i ; r_1 < \cdots < r_{d_i} \right\}$	d_i	Server
0.65	$\{6,8\}$	2	S_1
0.82	$\{1,5,9,10\}$	4	S_2
0.6	$\{4,7,15\}$	3	S_3
0.37	$\{12,14\}$	2	S_4
0.79	$\{2,3,11,13\}$	4	S_5

Table 4-Extracting absolute score of servers according to different values of K

ϕ_{ij} **for** $\beta(K) = \frac{K}{20} = 0.5$	ϕ_{ij} **for** $\beta(K) = \frac{K}{5} = 2$	$R_i = \left\{ r_k \mid k = 1, \ldots, d_i ; r_1 < \cdots < r_{d_i} \right\}$	d_i	Server S_i
0.76180	0.02778	$\{6,8\}$	2	S_1
2.09678	1.06234	$\{1,5,9,10\}$	4	S_2
1.13616	0.08735	$\{4,7,15\}$	3	S_3
0.55594	0.01204	$\{12,14\}$	2	S_4
1.86332	0.37529	$\{2,3,11,13\}$	4	S_5