SARIPOD: TOWARDS A MULTIAGENT POSSIBILISTIC SYSTEM FOR WEB INFORMATION RETRIEVAL

Bilel Elayeb, Montaceur Zaghdoud, Mohamed Ben Ahmed
RIADI-GDL Laboratory ENSI, Campus Universitaire 2010 la Manouba, Tunisia
{Bilel.Elayeb, Mohamed.Benahmed@riadi.rnu.tn}, Montaceur.Zaghdoud@ensi.rnu.tn

Fabrice Evrard
IRIT site ENSEEIHT, 02 Rue de Charles Camichel, 31071 Toulouse Cedex 7, France
Fabrice.Evrard@enseeiht.fr

Keywords: Multiagent System, Information Retrieval, Hierarchical Small-Worlds, Possibilistic Networks, Possibilistic relevance, user's preferences.

Abstract: We describe in this paper a multiagent possibilistic system for web information retrieval, called SARIPOD. This system is based on Hierarchical Small-Worlds (HSW) and Possibilistic Networks (PN). The first HSW consists in structuring the "Google" search results in dense zones of web pages which strongly depend on each other. We thus reveal dense clouds of pages which "speak" more or less about the same subject and which all strongly answer the user's query. The goal of the second HSW consists in considering the query as multiple in the sense that we don't seek only the keyword in the web pages but also its synonyms. The PN generates the mixing of these two HSW in order to organize the searched documents according to user's preferences.

1 INTRODUCTION

The key issue of Information Retrieval (IR) is that documents must be retrieved from a large document collection in response to a user's need, often on the basis of poor information. Known models in the literature (Boolean, vector space, probabilistic, Bayesian) represent documents and queries only through weighted lists of terms and a measure of relevance is computed (vector space similarity, probabilistic relevance) based on those weighted lists. Devising a proper weighting scheme seems to be the fundamental element of actual IR models since the computation of relevance relies on it (Ribeiro-Neto and al., 1996) (Sparck, 1998). Usually, the weighting scheme is the result of several combinations: term frequencies in document (tf), term frequencies in the whole collection (idf) and document length (dl) (Salton and al., 1994) (Singhal and al., 1996). Whatever the used model, the response to a user need is a list of documents ranked according to a unique relevance value. Many approaches consider term weights as a probability of relevance. In such models, the incompleteness of information is not considered when representing or evaluating documents given a query. Yet, the rough nature of document descriptions (a multiset of terms) and of the query description (a list of terms) are hardly compatible with the high precision of relevance values obtained by current methods.

The aim of this paper is to propose basic steps towards an IR mixed approach based on possibility and necessity measures. Instead of using a unique relevance value, we propose a possibilistic approach for computing relevance. This model should be able to infer propositions like: It is plausible with a certain degree that the document is relevant for the user need; It is almost certain (in possibilistic sense) that the document is relevant to the query; The set D_1 of documents (possibly singleton) is better than the set D_2 of documents.

The first kind of proposition is meant to eliminate irrelevant documents (weak plausibility). The second answer focuses attention on what looks very relevant. The third proposition suggests that, since the raw information on documents is more qualitative than quantitative, ordinal approaches to the problem may be interesting as well. The use of probability theory in the definition of relevance given a query does not account for our limited

knowledge of the relevance of a document, since it does not consider imprecision and vagueness intrinsic to relevance (Brini and al., 2003).

This paper is structured as follows: in the next section we present the hierarchical small-worlds graph. We briefly recall some notions of possibility theory in section 3. We describe in section 4 the multiagent architecture of SARIPOD system and we present the functionality of each agent. The experimentation of our system is in section 5 and it's evaluated in section 6. Section 7 suggests future works.

2 HSW GRAPH

Recent work in graph theory has revealed a set of features shared by many graphs observed "in the field". These features define the class of "hierarchical small-worlds" networks (Watts and Strogatz, 1998). The relevant features of a graph in this respect are the following (Newman, 2003):

D : the density of the network. HSWs typically have a low D, i.e. they have rather few edges compared to their number of vertices.

L : the average shortest path between two nodes. It is also low in a HSW.

C : the clustering rate. This is a measure of how often neighbours of a vertex are also connected in the graph. In a HSW, this feature is typically high.

I : the distribution of incidence degrees (i.e. the number of neighbours) of vertices according to the frequency of nodes. In a HSW network, this distribution follows a power law.

As a mean of comparison, table 1 (Gaume and al., 2004) shows the differences between random graphs (nodes are given, edges are drawn randomly between nodes), regular graphs and HSW. The graph of the web belongs to the class of HSW (Douglas and Houseman, 2002) (Sergi and Ricard, 2004).

Table 1: Comparing classes of graphs

with equal D	**L**	**C**	**I**
Random graphs	small L	Small C	Poisson Law
HSW	**small L**	**High C**	**power law**
Regular graphs	High L	High C	constant

3 POSSIBILISTIC LOGIC

Possibility theory introduced by (Zadeh, 1978) and developed by (Dubois and Prade, 1987), handles uncertainty in the interval [0,1] called possibility scale, in a qualitative or quantitative way.

3.1 Possibility distribution

Possibility theory is based on possibility distributions. The latter, denoted by π, are mappings from Ω (the universe of discourse) to the scale [0,1] encoding partial knowledge on the world. The possibility scale is interpreted in two ways. In the ordinal case, possibility values only reflect an ordering between possible states; in the numerical scale, possibility values often account for upper probability bounds (Dubois and Prade, 1998).

3.2 Possibility and necessity measures

A possibility distribution π on Ω enables events to be qualified in terms of their plausibility and their certainty, in terms of possibility and necessity measures respectively.

The possibility $\Pi(A) = \max_{x \in A} \pi(x)$ of an event A relies on the most normal situation in which A is true.

The necessity $N(A) = \min_{x \notin A}(1- \pi(x)) = 1-\Pi(\neg A)$ of an event A reflects the most normal situation in which A is false.

The width of the gap between N(A) and $\Pi(A)$ evaluates the amount of ignorance about A. Note that $N(A) > 0$ implies $\Pi(A) = 1$. When A is a fuzzy set this property no longer holds but the inequality $N(A) \leq \Pi(A)$ remains valid (Dubois and Prade, 1987).

3.3 Possibilistic Networks (PN)

A directed possibilistic network on a variable set V is characterized by a graphical component and a numeric component. The first one is a directed acyclic graph. The graph structure encodes independence relation sets just like Bayesian nets (Benferhat and al., 2002) (Borgelt and al., 2000). The second component quantifies distinct links of the graph and consists of the conditional possibility matrix of each node in the context of its parents. These possibility distributions should respect normalisation. For each variable V:

- If V is a root node and dom(V) the domain of V, the prior possibility of V should satisfy: $\max_{V \in dom(V)} \Pi(V) = 1$;
- If V is not a root node, the conditional distribution of V in the context of its parents context satisfy:
  $\max_{V \in dom(V)} \Pi(V \mid Par_V) = 1$;
  $Par_V \in dom(Par_V)$; where: dom(v): domain of V; Par_V : value of parents of V; $dom(Par_V)$: domain of parent set of V.

4 SARIPOD SYSTEM

The fact that we deals with sources of information collected from Internet network, made us choose the development of crawler agent able to crawl the Internet. It appeared also intuitive to us to interface the user by means of interface agents. Finally, the fact that we deals with open and dynamic environments made us choose the development of an intermediate layer of agents. We thus see appearing three levels of abstraction in the multiagent architecture of SARIPOD (see Figure 1).

4.1 User Agent

The user agent is the entry-gate of the external queries to the system. It provides to user the good form which will enable him to easily formulate a query. User's query is made up of the URL of the root web page as well as a set of required keywords. The user agent is perceptive and autonomous in the sense where it is able to keep user's preferences when this one uses the system. It is able to store information for the user and to act like a resource agent.

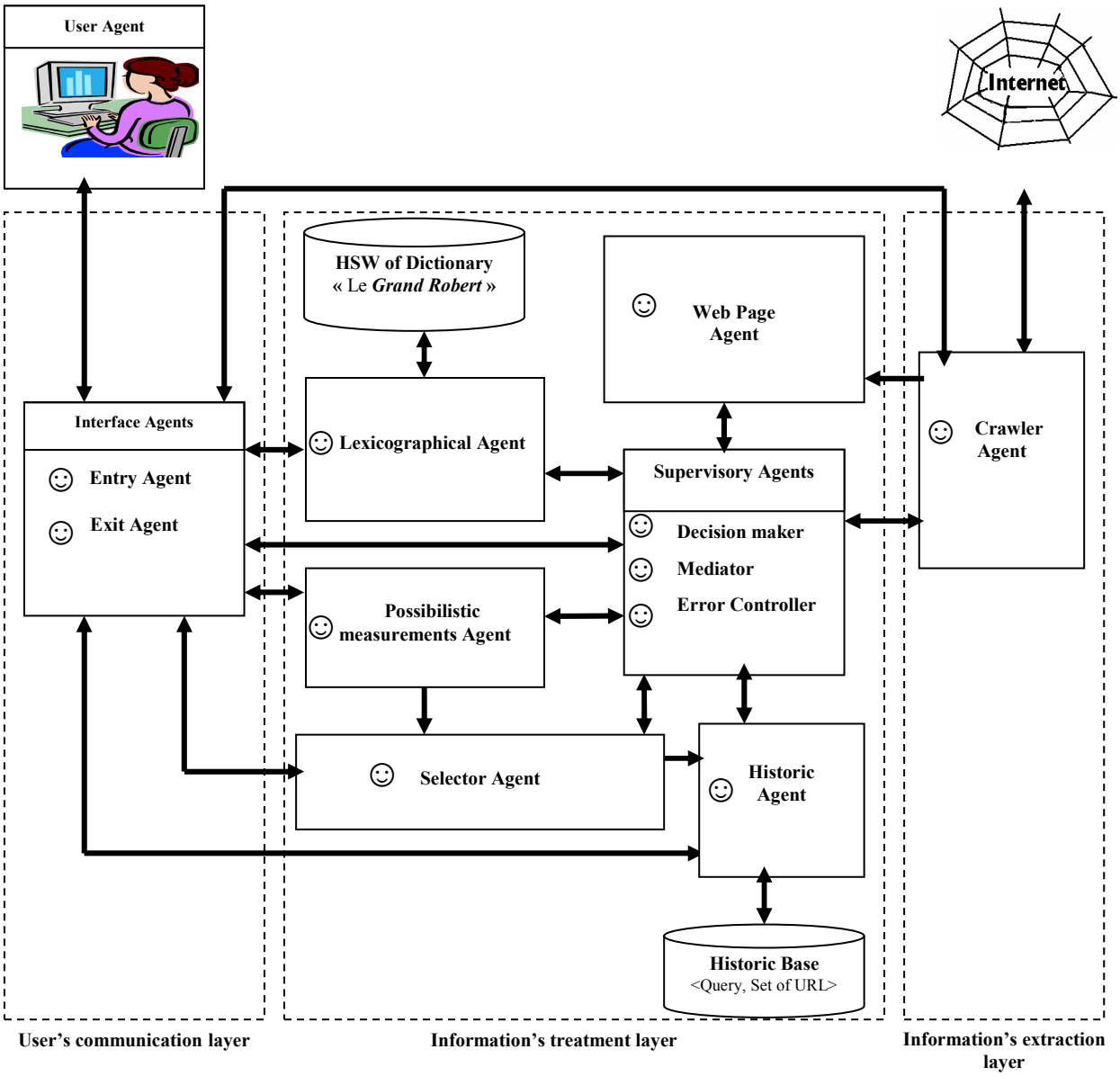

Figure 1: Multiagent architecture of SARIPOD system

4.2 Interface Agents

They ensure the communication between the system and its users. They are of two types: (1) *Entry Agent.* Analyzes user's query and transmits thereafter the keywords sought to the lexicographical agent which determines their synonyms starting from the HSW of dictionary of words. (2) *Exit Agent.* Is charged to present the results of search at the user, from where the "adaptive" term; it is able to adapt the results of search to user's preferences.

4.3 Supervisory Agents

They take care of the correct operation of the system; all other agents must be with their service and under their responsibility. They are charged to assign the tasks of research of information process to the various agents, to decide in the event of a multitude of choice and to control the possible errors at a session of selection of the most relevant web documents.

4.3.1 Mediator Agent

The mediator agent plans the various tasks of search of information and assigns them to various agents of the system, it is a driving role which can easily narrow where the system becomes completely distributed; i.e. it is inversely proportional to the degree of cognition of the other agents of the system. In this first version of SARIPOD system, the mediator-facilitator agent plays the role of a facilitator (Ferber, 1995).

4.3.2 Decision maker Agent

This decision maker agent has a fundamental role in SARIPOD system. Initially, it is charged to make a post-processing selection after carrying out the various selected web pages by the selector agent so that the exit agent knows organized this result in the order preferred by the user.

4.3.3 Error Agent Controller

It is charged to control the correct operation of the system by carrying out the directives of control of the errors communicated by each agent of the system. It informs the decision maker agent of what occurs in the system, which in its turn decides to stop or not an agent. Often, it analyzes the cause of error of each agent in difficulty, if it is for example about a lack of information, he tries to solve this problem by asking more information from the agent source of error. In the worst of case, it decides to stop the operation of an agent.

4.4 Lexicographical Agent

The lexicographical agent is interested in the selection of synonymy through the examination of the dictionary graph of words. Our approach consists in representing the dictionary by an HSW graph : there is an arc of a top A towards a top B if and only if the entry B appears in the definition of entry A as a synonym (Awada and Chebaro, 2004).
The data base of this agent is the French dictionary "Le Grand Robert" with XML format in which the elements are described by a whole of beacons allowing each one to associate semantics the various components.

4.5 Crawler Agent

Crawler agent (Miller and Bharat, 1998) is based on our "Strat" algorithm of crawling. The input of this agent is the reformulated query and it's output the HSW of web pages and a set of their URLs.
We propose within this framework a systematic crawling technique via "Strat" algorithm, whose scenario is as follows:

1. While a page on N contains the word *w* it is necessary to continue to visit the outgoing pages of this page;
2. When N successive pages do not contain the word *w* (whatever the depth), we stop research in this branch, so we need backtrack then.

4.6 Web page Agent

This agent allows the extraction of the logical structure of each web document. We mind to store such a document in an editable and exchangeable format that represents explicitly its structure and its content. The strategy of this agent is based on a labelling method. It is composed of several analysis steps that leads to the transformation of the document in a logical structure where each text block has a level and a label that represents explicitly its logical role (Bounhas and al., 2007).

4.7 Possibilistic measurements Agent

The aim of this agent is to propose basic steps towards an IR mixed approach based on possibility and necessity measures. Instead of using a unique relevance value, we propose a Possibilistic approach

for computing relevance. This agent should be able to infer propositions like:

- It is plausible to a certain degree that the document is relevant for the user need.
- It is almost certain (in possibilistic sense) that the document is relevant to the query.

The first kind of proposition is meant to eliminate irrelevant documents (weak plausibility). The second answer focuses attention on what looks very relevant (Brini and al., 2003). This agent encodes relationship dependencies existing between query terms (lexicographical agent) and web documents (web page agent) through naïve possibilistic networks and quantifies these relationships by two measures: possibility and necessity. This agent allotted a coefficient of relevance to each logical entity according to its importance in the web document. These coefficients are calculated according to the following way:

$\alpha_{ML} = ML + Max(\alpha_{Legends}, \alpha_{Paragraph})$ (1)

$\alpha_{Li} = ML - L_i + Max(\alpha_{Legends}, \alpha_{Paragraph})$ (2)

Where ML is the maximal level, L_i is the level *I* of logical entity.

The quantitative relevance of each logical entity of a web document of the collection, with the query is Q = $(t_1, t_2,\ldots, t_T)$, is calculated in the following way:

The expression of $\Pi(LEd_j|Q)$ is then proportional to:

$$\Pi'(LEd_j|Q) = \Pi(t_1|LEd_j)*\ldots* \Pi(t_T|LEd_j) = nft_{1j} *\ldots* nft_{Tj} \quad (3)$$

Where $nft_{ij} = tf_{ij}/\max(tf_{kj})$: the normalized frequency of the terms of the query in the logical entity.

The certainty to restore a logical entity (LE) of a relevant document d_j for a query, noted N(LEdj|Q), is given by:

$N(LEd_j|Q) = 1- \Pi(\neg LEdj|Q)$ (4)

Where:

$\Pi(\neg LEd_j|Q) = (\Pi(Q|\neg LEd_j)* \Pi(\neg LEd_j))/\Pi(Q)$ (5)

In the same way, $\Pi(\neg LEd_j|Q)$ is then proportional to:

$\Pi'(\neg LEd_j|Q) = \Pi(t_1| \neg LEd_j)* \ldots*\Pi(t_T|\neg LEd_j)$ (6)

This numerator can be expressed by:

$\Pi'(\neg LEd_j|Q) = (1- \phi LE_{1j})*\ldots* (1- \phi LE_{Tj})$ (7)

$\phi LE_{ij} = Log_{10}(nCLE/nLEd_i)*(nft_{ij})$ (8)

And : nCLE = The number of logical entity of the documents of the collection,

$nLEd_i$ = The number of logical entity of the documents of the collection, containing the term t_i.

Let us note the degree of relevance mixed possibilistic of logical entity of the document d_i by:

$DRMPLE_j(d_i) = \Pi(LEd_i|Q) + N(LEd_i|Q)$ (9)

We note finally the degree of relevance mixed possibilistic of the document d_i by:

$DRMP(d_i) = \sum_j(\alpha_j * DRMPLE_j(d_i))$ (10)

User's preferences of SARIPOD system are defined as the quality of the document which he seeks; i.e. his preferences for certain stylistics attributes in the searched documents: information located either in the principal title of the document, or in the sub-titles, or in the paragraphs... and also his preferences for certain types of information: information in figures, tables or multimedia sequences.

The preferred documents are those which have a high value of $DRMP(d_i)$. Let us note that the α_j are parameterized in our system and can be modified according to the user's preferences.

Example: Assume a three documents collection containing the four terms t_1, t_2, t_3 and t_4:

$d_1 = \{t_1, t_1, t_1, t_2, t_2, t_3\}$, $d_2 = \{t_1, t_1, t_2, t_2, t_2, t_2\}$, $d_3 = \{t_1, t_3, t_3, t_3, t_3, t_4, t_4\}$

These terms are distributed in the logical entities of these three documents as table 2 indicates. The degree of relevance mixed possibilistic of each document d_i is $DRMP(d_i)$.

Table 2: Distribution of the terms in the logical structures of the three documents

Logical structure of document	**d_1**	**d_2**	**d_3**
Maximal Level (ML)	t_1	t_1, t_2	t_4
ML-1	t_2		t_1, t_3
ML-2			t_3
ML-3		t_2	
ML-4	t_3		
Figure Legend (FL)			t_3
Table Legend (TL)	t_2		
Multimedia Sequence Legend (MSL)		t_2	
Paragraph (P)	t_1, t_1	t_1, t_2	t_3, t_4
$DRMP(d_i)$	**11.28**	**11.76**	**19.07**

For the example of this query Q, the document d_3 is more preferred than documents d_2 and d_1. We notice that the most relevant document is the one whose query's terms exist in its logical entities having the significant coefficients of relevance α_j, such as the maximum level (e.g., the principal title of the document), etc. Thanks to our mixed possibilistic approach, we also noticed that: even if the selected terms tend to select this document, these terms are not most frequent of the document (t_4 isn't the most frequent term of d_3).

4.8 Selector Agent

This agent should be able to infer propositions like: document d_1 is more appropriate than document d_2 or the set $\{d_1, d_2\}$ is better than the set $\{d_3,d_4\}$.

Indeed, this proposition suggests that, since the raw information on documents is more qualitative than quantitative, ordinal approaches to the problem may be interesting as well (Brini and al., 2004). This agent sorts web documents in a descending order of their degrees of possibilistic relevance (DRMP). The document which more answering user's preferences will be posted at the head of the sorted list of documents, turned over to the exit agent, which checks its conformity with user's preferences.

4.9 Historic Agent

This agent makes it possible to build a base of history of the queries and their answers, already passed by the system. In the reception of a new query, the system consults this base of history, seeks the nearest query in this base, using Case Base Reasoning (CBR) technique (Berry and Linof, 1997) and finally, it updates the answer by eliminating URLs that are not available on the web and by adding nonexistent new URLs in this base of history.

5 EXPERIMENTATION

We made the tests of SARIPOD system (see figure 2) on a local site containing 10000 pages and for keywords having a variable number of synonyms. Table 3 gives our preliminary results. Parameters L and C show the HSW structure of the web.
We notice that the difference between the degree of possibilistic relevance of the most relevant page (DRMP (d_1)) and of the least relevant page of the collection (DRMP (d_N)) decreases when the number of selected URL increases. It proves that the first goal which motivated us for the use of the HSW is checked here: the change of answers to research query given by *Google* web search engine, by structuring it in a HSW so that, if a page among the returned answers seems relevant then all neighbours in this HSW will be too. So, we increase the number of Google's returned documents and we consequently change the Google's PageRank (Vise and Malseed, 2006).

6 EVALUATION

As evaluation of the SARIPOD system we present on one hand a justification of the use of the HSW graph in information retrieval, and on the other hand its evaluation compared to the traditional web search engine systems such as Google.
Indeed, we distinguish two very significant uses of these two HSW and their mixing in SARIPOD system: The first use consists in structuring the "Google" search results in dense zones of web pages which strongly depend on each other. We thus reveal dense clouds of pages which "speak" more or less about the same subject and which all strongly answer the user's query. For another cloud of web pages strongly related to each other, it is similar: all of them answer this same query. The essential difference is that each cloud of web pages strongly answers the query in a particular way.

Table 3: Experimentations results

Number of synonyms	Required keywords	Numbers of URLs obtained	L	C	Duration of query (in second)	Degree of relevance of doc. 1 DRMP (d_1)	Degree of relevance of doc. N DRMP (d_N)
0	vérifier	61	1,0690	0,4499	15	39,22	7,33
1	vérifier examiner	93	1,3305	0,3749	17	29,74	5,77
2	vérifier examiner voir	207	1,1167	0,3607	20	16,23	3,69
3	vérifier examiner voir éprouver	363	1,2491	0,2477	21	9,53	2,87
4	vérifier examiner voir éprouver reconnaître	412	1,0148	0,3258	24	7,19	1,88
5	vérifier examiner voir éprouver reconnaître essayer	517	1,4124	0,3281	29	5,37	1,45
6	vérifier examiner voir éprouver reconnaître essayer contrôler	761	1,3215	0,3562	35	3,86	0,83
7	vérifier examiner voir éprouver reconnaître essayer contrôler expérimenter	833	1,3685	0,3421	42	1,34	0,67
8	vérifier examiner voir éprouver reconnaître essayer contrôler expérimenter constater	904	1,3724	0,3212	55	0,53	0,18

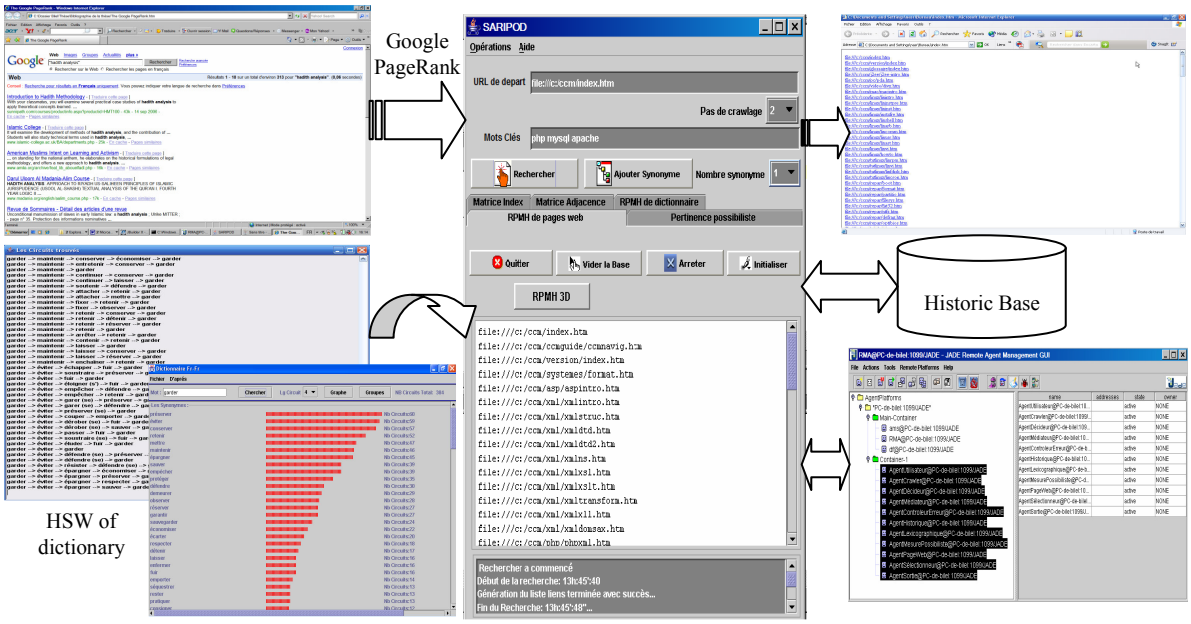

Figure 2: Overview of SARIPOD system: input, output and control interfaces

For example, the query "jouer", in the HSW of French synonyms, gives four clouds of verbs close to "jouer": the first cloud concerns A = {parier, risquer, miser, hasarder,...}, the second B = {tromper, mystifier, abuser, berner,...}, etc. for the two others. For the web, it's the same thing; a query (expressed with some keywords) returns a set of web pages (Google answers) which it's necessary to organize in HSW to reveal some large clouds of web pages among all these answers. Each cloud gathers a batch of pages which answer the query in relevant ways: as A pertinently answers the query "jouer" if A is interested in the "pari", as B which also answers pertinently the same query "jouer" if B is interested in the "abus", etc. For the web each cloud of web pages will be relevant and, thanks to additional keywords, it will be possible to select a particular cloud.

Quality lies in the fact that when we look at the web pages of the same cloud, all the pages are relevant, but if this degree is not yet sufficient, we can only make queries in this only cloud (contrary to Google which never organizes its 300.000 answers in clouds) to obtain a subset of web pages which we can again (thus recursively) organize in under-HSW. With the deepest of this structure we find web pages alone. The set of answers was thus organized in HSW and under-HSW to constitute a kind of decision tree (or structure of classification) on web pages according to the used keywords. Google can't do the same thing, but it can only search again in the set of preceding answers. In fact, Google is able to return web pages which our system would have put them in different clouds since the first query.

The second very significant use of the HSW consists in not taking the keywords just as they are but regarding a query as multiple in the sense that we don't search only the keyword in the web pages but also its synonyms. In fact, beyond strict synonymy, we will search for this keyword but also words close to it. The proximity of two words relies on circuits in the dictionary HSW. The words considered as nearly relations thus include the synonyms of this word but don't narrow down to them. There will be potentially (in practice that will be limited by a terminal) all the words more or less near query's keyword. This number of words is skeletal (1, 5, 100...). A query is thus now very flexible since it tolerates that a web page is a good answer even if it doesn't contain the searched keyword.

However to be able to have this flexibility we need obviously a dictionary and especially to have structured this dictionary (all its entries) in HSW to precisely know which word is near to which other. However there are many ways of emerging a structure of HSW starting from a dictionary (that of (Gaume and al., 2004) for example consists in using words' definitions: The word w_1 is connected to the word w_2 if and only if w_2 belongs to the definition of

w_1, using this relation he deduces a "semantic proximity" from any word to any other). We take again this definition and we calculate the proximity between the words in order to make the query more flexible. We can quantify from there the web pages obtained following a query using certain keywords. Each answer page will be characterized by a degree of relevance which will result from the combination of the degrees of proximity between the query's keywords and the words effectively present in this page.

7. CONCLUSION

This paper presents a web information retrieval system based on Hierarchical Small-Worlds (HSW) and Possibilistic Networks (PN). The first HSW consists in structuring the "Google" search results in dense zones of web pages which strongly depend on each other. We thus reveal dense clouds of pages which "speak" more or less about the same subject and which all strongly answer the user's query. The goal of the second HSW consists in considering the query as multiple in the sense that we don't seek only the keyword in the web pages but also its synonyms. The PN generates the mixing of these two HSW in order to organize the searched documents according to user's preferences.

The approach proposed by Brini and al. (Brini and al., 2004) is only based on the quantitative setting of possibility calculus; our mixed approach extends it by also focusing on the ordinal setting. In fact, we add a qualitative approach which introduces preferences between retrieved documents. This system had been implemented within Java language and Jade multiagent platform.

We plan to carry out finer measurements of the performances of SARIPOD system by extending the tests with other types of web documents and by giving preferences between query terms.

References

Awada, A., Chebaro, B., 2004. Etude de la synonymie par l'extraction de composantes N-connexes dans les graphes de dictionnaires. In *JEL2004*, Nantes, France.

Benferhat, S., Dubois, D., Garcia, L., Prade, H., 2002. On the transformation between possibilistic logic bases and possibilistic causal networks. In *International Journal of Approximate Reasoning*, 29, n°2, 135-173.

Berry, M., J., A., Linof, G., 1997. *Data Mining : Techniques appliquées au marketing, à la vente, et aux services clients*. InterEditions, Paris.

Borgelt, C., Gebhardt, J., Kruse, R., 2000. Possibilistic Graphical Models. *Computational Intelligence in Data Mining, CISM Courses and Lectures* 408, 51- 68.

Bounhas, I., Habacha, A., Ben Ahmed, M., 2007. Labelling Based Approach for Logical Structuration of Scientific Web papers. *In The ICSOFT 2007,* Bercelona, Spain.

Brini, A., H., Boughanem, M., Dubois, D., 2004. Towards a possibilistic approach for information retrieval. *Data and Knowledge Engineering Proceedings EUROFUSE 2004*, Varsovie, Pologne, p. 92-102.

Brini, A., H., Boughanem, M., 2003. Relevance feedback : introduction of partial assessments for query expansion. *Proc. 2d. EUSFLAT Conf*, Zittau, Allemagne, 67-72.

Douglas, W., Houseman, M., 2002. The navigability of strong ties: small worlds, ties strength and network topology. *eScholarship Repository*, University of California.

Dubois, D., Prade, H., 1998. Possibility Theory : Qualitative and Quantitative Aspects. In *Handbook on Defeasible Reasoning and Uncertainty Management Systems*, vol.1, Kluwer , 21-42.

Dubois, D., Prade, H., 1987. *Théorie des Possibilités : Application à la Représentation des Connaissances en Informatique*. Edition MASSON. Paris.

Ferber, J., 1995. *Les systèmes multi-agents, vers une intelligence collective*. InterEditions, Paris.

Gaume, B., Hathout, N., Muller, P., 2004. Désambiguïsation par proximité structurelle. *TALN 2004*, Fès, Maroc.

Miller, R., C., Bharat, K., 1998. SPHINX : A framework for creating personal, site-specific web crawlers. *In the WWW7*. Printed in Computer Network and ISDN System v.30, pp. 119-130, Brisbane, Australia.

Newman, M., E., J., 2003. The structure and function of complex networks. *SIAM Review*, volume 45, 167–256.

Ribeiro-Neto, B., Silva, I., Muntz, R., 1996. A Belief Network Model for IR. *Proc. Of the 19th ACM SIGIR Conf. on Research and Development in Information Retrieval*, 253-260.

Salton, G., Allan, J., Buckley, C., Singhal, A., 1994. Automatic Analysis, Theme Generation and Summarization of Machine Readable Texts. *Science*, 264, 3, 1421-1426.

Sergi, V., Ricard, V., S., 2004. Hierarchical Small Worlds in Software Architecture. *IEEE transaction on Software Engineering*. USA.

Singhal, A., Salton, G., Mitra, M., Buckley, C., 1996. Document Length Normalization. *Information Processing and Management*. 32(5) : 619-633.

Sparck, J., K., 1998. A Look Back and a Look Forward. *Proceedings of the 11th annual international ACM SIGIR Conf. on Research and Development in Information Retrieval*, 13-29.

Vise, D., Malseed, M., 2006. *Google story*, Edition Dunod, Paris.

Watts, D., Strogatz, S., 1998. Collective dynamics of 'small-world' networks. *Nature*, (393), 440–442.

Zadeh, L., A., 1978. Fuzzy Sets as a basis for a theory of Possibility. *Fuzzy Sets and Systems*. 1 :3-28.