Identification number: DOI: 10.1108/17415650910965191
Official URL: http://dx.doi.org/10.1108/17415650910965191

Towards an intelligent possibilistic web information retrieval using multiagent system

Bilel Elayeb
Laboratoire RIADI-GDL, Ecole Nationale des Sciences de l'Informatique (ENSI), Université de la Manouba, Manouba, Tunisia, and Institut de Recherche en Informatique de Toulouse (IRIT), Toulouse, France

Fabrice Evrard
Institut de Recherche en Informatique de Toulouse (IRIT), Toulouse, France, and

Montaceur Zaghdoud and Mohamed Ben Ahmed
Laboratoire RIADI-GDL, Ecole Nationale des Sciences de l'Informatique (ENSI), Université de la Manouba, Manouba, Tunisia

Abstract

Purpose – The purpose of this paper is to make a scientific contribution to web information retrieval (IR).

Design/methodology/approach – A multiagent system for web IR is proposed based on new technologies: Hierarchical Small-Worlds (HSW) and Possibilistic Networks (PN). This system is based on a possibilistic qualitative approach which extends the quantitative one.

Findings – The paper finds that the relevance of the order of documents changes while passing from a profile to another. Even if the selected terms tend to select the relevant document, these terms are not the most frequent of the document. This criterion shows the asset of the qualitative approach of the SARIPOD system in the selection of relevant documents. The insertion of the factors of preference between query terms in the calculations of the possibility and the necessity consists in increasing the scores of possibilistic relevance of the documents containing these terms with an aim of penalizing the scores of relevance of the documents not containing them. The penalization and the increase in the scores are proportional to the capacity of the terms to discriminate between the documents of the collection.

Research limitations/implications – It is planned to extend the tests of the SARIPOD system to other grammatical categories, like refining the approach for the substantives by considering for example, the verbal occurrences in names definitions, etc. Also, it is planned to carry out finer measurements of the performances of SARIPOD system by extending the tests with other types of web documents.

Practical implications – The system can be useful to help research students find their relevant scientific papers. It must be located in the document server of any research laboratory.

Originality/value – The paper presents SARIPOD, a new qualitative possibilistic model for web IR using multiagent system.

Keywords Information systems, User studies, Internet, Worldwide web

Paper type Research paper

1. Introduction

When we make a web search query on the internet using keywords (thanks to a search engine like Google), we obtain in general a considerable list of links of web pages answering this query. We can also, by using complementary keywords, seek something more relevant in the whole of the preceding results. And so on until obtaining required result. However, we could differently proceed by structuring the whole of the web

DOI 10.1108/17415650910965191

pages obtained in the first result. The structure would consist in classifying all these results by domains and sub-domains. A very promising technique emerges today and called upon the hierarchical small-worlds. Thus, each web page would be a node of a gigantic graph whose edges would be the hypertextual links of a page towards another. Certain calculations on this graph make regroupings sets of web pages which "speak" about almost the same subject.

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 *et 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 *et al.*, 1994; Singhal *et 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 qualitative 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 Boughanem, 2003).

We present, in this framework, a multiagent system for web IR, 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. Indeed, SARIPOD is a new approach for IR model based on possibility and necessity measures. This model encodes relationship dependencies existing between query terms and web documents through naïve PN and quantifies these relationships by two measures: possibility and necessity. The retrieved documents are those which are necessarily or possibly relevant given a user's query. The search process restores the plausibly or necessarily relevant documents for a user need. The user's query is seen like new information to propagate in a PN. Moreover, if the basic approach takes account of the quantitative aspect here, our system extends it to the qualitative possibilistic framework.

This paper is structured as follows: in the next section we present the HSW 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. Section 5 presents the importance of qualitative approach in SARIPOD system. We introduce preferences between query's terms in SARIPOD system in section 6. The experimentation of our system is in section 7. We present a synthesis and discussion of the system in section 8. Section 9 suggests future works.

2. Hierarchical small-worlds

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 neighbors 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 neighbors) of vertices according to the frequency of nodes (how many nodes are there that have an incidence degree of 1, 2, ..., *n*). In a HSW network, this distribution follows a power law (Douglas and Houseman, 2002) (Gaume *et al.*, 2004) (Sergi and Ricard, 2004).

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

A directed PN 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 (Borgelt *et al.*, 2000; Benferhat *et al.*, 2002). 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: $\max_{V \in \mathrm{dom}(V)} \Pi(V) = 1$.
- If V is not a root node, the conditional distribution of V in the context of its parents context satisfy: $\max_{V \in \mathrm{dom}(V)} \Pi(V | \mathrm{Par}_V) = 1$.
- $\mathrm{Par}_V \in \mathrm{dom}(\mathrm{Par}_V)$; 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 deal with sources of information collected from internet network, made us choose the development of crawler agent able to explore (crawl) the internet. It appeared also intuitive to us to interface the user by means of interface agents. Finally, the fact that we deal 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.

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.

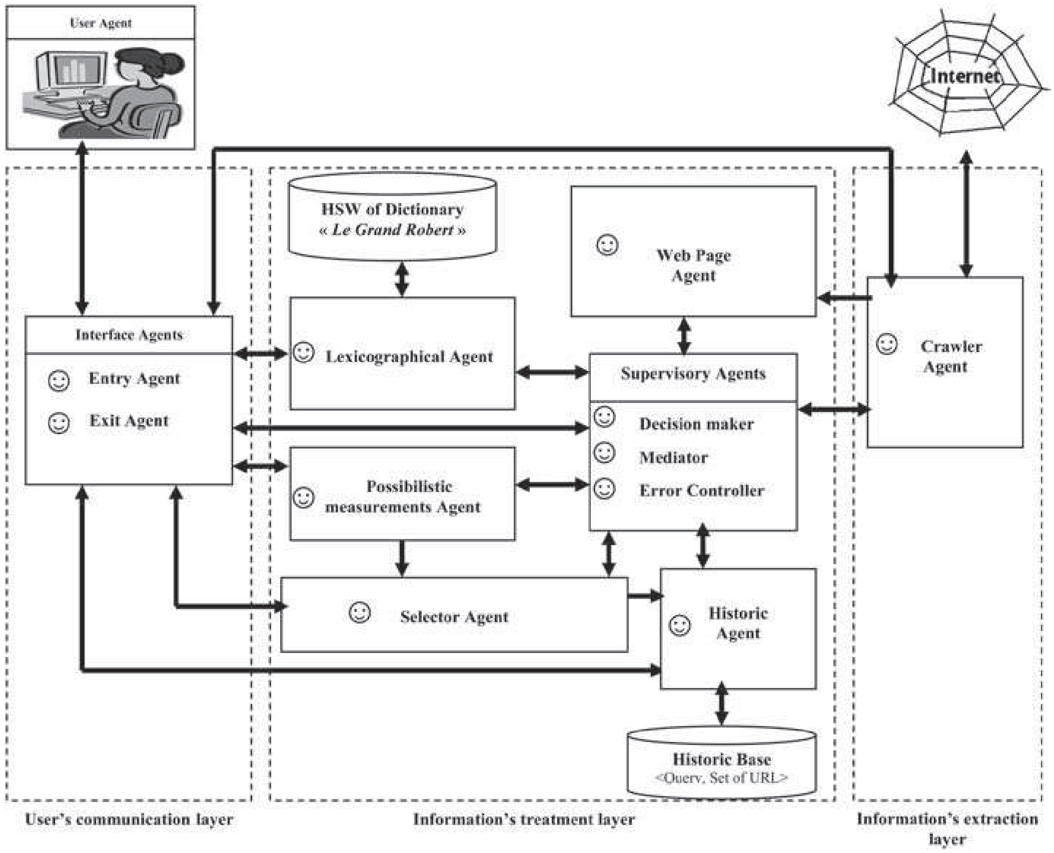

Figure 1.
Multiagent architecture of SARIPOD system

(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 (Zaghdoud, 2003).

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 caring 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. In addition, soon, this agent will be equipped with an exceptional intelligence to make a pre-processing choice of the relevant web documents, thus enabling him to earn the system a considerable time.

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; Awada, 2005).

This agent calculates a new proximity between two words w_1 and w_2 in term of the number of circuits passing through w_1 and w_2 and ghost in w_1 in the following way:

$$\text{Proximity\_dictionary}(w_1, w_2) = \frac{\text{The number of circuits } (w_1, w_2)}{\text{The maximum number of detected circuits}}.$$

The database 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. The enormous volume of corresponding XML file has constrained us to reduce the size of this dictionary by keeping only information which interests us to make the treatment more effective. This was done, obviously, without loss of relevant information. In fact, we limit ourselves only to the nouns, verbs, adjectives and adverbs of this dictionary, which exempt to us blank words such as pronouns, definite and indefinite articles, prepositions and auxiliaries (to be and to have).

4.5 Crawler agent

A web crawler is a program which automatically traverses the web by downloading documents and following links from page to page. It starts from a starting URL of a web page and it has a changeable depth of propagation (Miller and Bharat, 1998). Crawler agent is based on our "Strat$_\delta$" algorithm of crawling. The input of this agent is the reformulated query and its output the HSW of web pages and a set of their URLs.

We propose within this framework a systematic crawling technique via "Strat$_\delta$" algorithm, whose scenario is as follows:

(1) We keep any page P which contains a word V sufficiently close to word W with the direction where:

$$\pi = \text{Proximity\_dictionary}(W, V) \geq \delta.$$

(2) We explore the outgoing pages of P until a depth limits N (increasing function of Proximity_dictionary (W, V) for V of this page, but raised function nevertheless).

(3) The limit of exploration is updated by that of the deepest page of the explored branch.

The first stage (1) makes it possible to keep pages which contain the word W or its close if $\pi \geq \delta$ and δ is quite selected. The second (2) ensures that if we have $\pi < \delta$ we continue the exploration of as much less far π is small and we do not continue in any event not indefinitely. The third (3) makes it possible to set out again of more beautiful if at the end of pages not very interesting we fall down on a page concerned strongly with W (or its close).

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 lead 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. Searching the user's keywords in such logical entity (LE) of document and not in the whole document proves the qualitative character of SARIPOD system.

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 Boughanem, 2003).

This agent encodes relationship dependencies existing between query terms (lexicographical agent) and web documents (web page agent) through naïve PN and quantifies these relationships by two measures: possibility and necessity. The retrieved documents are those which are necessarily or possibly relevant given a user's query. The search process restores the plausibly or necessarily relevant documents for a user need. This agent allotted a coefficient of relevance to each LE according to its importance in the web document. These coefficients are calculated according to the following way:

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

$$\alpha_{\mathrm{L}i} = \mathrm{ML} - L_i + \mathrm{Max}(\alpha_{\mathrm{Legends}}, \alpha_{\mathrm{Paragraph}}) \tag{2}$$

where ML is the maximal level, L_i is the level i of LE.

The quantitative relevance of each LE of a web document of the collection, with the query is $Q = (t_1, t_2, \ldots, t_{\mathrm{T}})$, is calculated in the following way.

The expression of $\Pi(\mathrm{LEd}_j \mid Q)$ is then proportional to:

$$\Pi'(\mathrm{LEd}_j|Q) = \Pi(t_1|\mathrm{LEd}_j) * \ldots * \Pi(t_{\mathrm{T}}|\mathrm{LEd}_j) = \mathrm{nft}_{1j} * \ldots * \mathrm{nft}_{\mathrm{T}j} \tag{3}$$

where $\mathrm{nft}_{ij} = \mathrm{tf}_{ij}/\max(\mathrm{tf}_{kj})$: the normalized frequency of the terms of the query in the LE.

The certainty to restore a LE of a relevant document d_j for a query, noted $N(\mathrm{LEd}_j \mid Q)$, is given by:

$$N(\mathrm{LEd}_j|Q) = 1 - \Pi(\neg\mathrm{LEd}_j|Q) \tag{4}$$

where:

$$\Pi(\neg\mathrm{LEd}_j|Q) = (\Pi(Q|\neg\mathrm{LEd}_j) * \Pi(\neg\mathrm{LEd}_j))/\Pi(Q). \tag{5}$$

In the same way, $\Pi(\neg\mathrm{LEd}_j \mid Q)$ is then proportional to:

$$\Pi'(\neg\mathrm{LEd}_j|Q) = \Pi(t_1|\neg\mathrm{LEd}_j) * \ldots * \Pi(t_\mathrm{T}|\neg\mathrm{LEd}_j). \tag{6}$$

This numerator can be expressed by:

$$\Pi'(\neg\mathrm{LEd}_j|Q) = (1 - \phi\mathrm{LE}_{1j}) * \ldots *(1 - \phi\mathrm{LE}_{\mathrm{T}j}) \tag{7}$$

$$\phi\mathrm{LE}_{ij} = \mathrm{Log}_{10}(n\mathrm{CLE}/n\mathrm{LEd}_i)*(\mathrm{nft}_{ij}) \tag{8}$$

and:

- $n\mathrm{CLE} =$ the number of LE of the documents of the collection; and
- $n\mathrm{LEd}_i =$ the number of LE of the documents of the collection, containing the term t_i.

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

$$\mathrm{DRMPLE}_j(d_i) = \Pi(\mathrm{LEd}_i|Q) + \mathrm{N}(\mathrm{LEd}_i|Q). \tag{9}$$

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

$$\mathrm{DRMP}(d_i) = \Sigma_j(\alpha_j * \mathrm{DRMPLE}_j(d_i)). \tag{10}$$

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. Indeed, if we seek, for example, the documents containing the word "*w*" in figures, it is enough to give the greatest importance to the coefficient of relevance corresponding to the figure legend (α_{FL}). Consequently, DRMP(d_i) of these documents will be most significant and will be posted in the heading of the list result of sorted documents.

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 Boughanem, 2003). 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

The historic agent of SARIPOD allows its users a significant profit in term of response time of our system. Indeed, 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 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.

Thus, the principal task of this agent consists in adding to the system a certain aptitude of training thus enabling it to benefit from the already played queries, for classes of a given user. Indeed, the system will be able to improve the user's profile.

5. The importance of qualitative approach in saripod system

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 I indicates. The degree of relevance mixed possibilistic of each document d_i is DRMP(d_i). Let us recall that the approach of (Brini *et al.*, 2005) is a quantitative approach which not taking account of the site of query's terms in the logical entities of the documents of the collection. We note DRP(d_i), the degree of relevance possibilistic of each document d_i calculated by this approach. User's profiles are given in Table II. The evaluation of the documents d_1, d_2 and d_3 for the query $Q = (t_1, t_2, t_3, t_4)$ is given in Table III.

The query Q interpreted as a conjunction of terms is too restrictive since no document contains all query terms simultaneously. Thus, necessity and possibility degrees of the documents equal 0. To avoid such case, we retrieve documents contained at least two terms and, if not productive, at least one term. If the query does not involve enough terms, the possibility of documents is equal 1 and their necessity 0. We then seek the documents which treat sets $\{t_1, t_2\}$ or $\{t_1, t_4\}$, or $\{t_2, t_4\}$. We see through this example, the need for allowing the user to express preferences between the query terms (see section 6).

In addition, we notice that our approach is finer than the quantitative one (Brini *et al.*, 2005) in the calculation of the possibilistic relevance of documents of the

Logical structure of document	d_1	d_2	d_3
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		
FL			t_3
TL	t_2		
MSL		t_2	
P	t_1, t_1	t_1, t_2	t_3, t_4

Table I. Distribution of the terms in the logical structures of the three documents

Logical structure of document	Coefficients α_j Profile1 (P1)	User's profiles Coefficients α_j Profile2 (P2)	Coefficients α_j Profile3 (P3)
ML	10	2	2
ML-1	9	5	8
ML-2	8	6	6
ML-3	7	9	4
ML-4	6	7	9
FL	5	3	7
TL	4	4	10
MSL	3	10	5
P	2	8	3

Table II.
The three user's profiles

	Profile1 (P1)			Profile2 (P2)			Profile3 (P3)		
	d_1	d_2	d_3	d_1	d_2	d_3	d_1	d_2	d_3
Brini *et al.* (2005):									
DRP(d_i)	0.16	0.18	0.24	0.16	0.18	0.24	0.16	0.18	0,24
Document's relevance order	3	2	1	3	2	1	3	2	1
SARIPOD: DRMP(d_i)	11.28	11.76	19.07	9.48	14.92	14.77	13.86	7.22	15.23
Document's relevance order	3	2	1	3	1	2	2	3	1

Table III.
Results of qualitative approach of SARIPOD system

collection, because we contributed to increase the relevance's scores of the documents containing these terms with an aim of penalizing the relevance's scores of the documents not containing them.

Let us note that the scores of the possibilistic relevance of the three documents, calculated by the approach of (Brini *et al.*, 2005) are very weak compared to those calculated by our approach and this is thanks to the paramount factors in our qualitative approach which is noted the coefficients of relevance α_j. Indeed, for the first user's profile (P1), the differences between scores are weak in the first approach (0.02; 0.08 and 0.06) because there are almost the same numbers of query's terms appearing in the three document (6; 6 and 7), whereas in the case of our approach, these differences between scores are much more remarkable (0.48; 7.79 and 7.31), which shows well the difference between a relevant document compared to those which are less relevant in the collection.

For the example of this query Q and for certain profiles, the relevance's order of documents changes by changing user's profile. Indeed, for the first profile (P1), document d_3 is more preferred than documents d_2 and d_1 in the both approach. This is due to the number of terms appearing in d_3 on the one hand (for the two approaches), and to term t_4 which appears in a LE having a significant weight (for our approach). Whereas, for the both other profiles (P2 and P3), the relevance's order of documents changes compared to the first profile (P1). In fact, and according to our approach, 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 maximum level (ML) and (ML-1) for the first profile, Multimedia Sequence Legend (MSL) MSL and (ML-3) for the second profile, Table Legend (TL) and (ML-4) for the third profile, etc. (see Table III).

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). This criterion shows the asset of the qualitative approach of SARIPOD system in the selection of relevant documents.

6. Preferences between query's terms in saripod system

Let us consider a query $Q(t_1, t_2, t_3)$ composed of three terms. It will be, after reformulation, the query $Q'(t_1, t_{11}, t_{12}, t_{13}, t_2, t_3, t_{31}, t_{32})$. With, t_{11}, t_{12}, t_{13} are the three words close to t_1 and t_{31}, t_{32} are the two words close to t_3 chosen by the user of the system. In fact, these close terms are automatically inserted in Q' each time the user seizes a number of terms close for a given term of the query Q.

We define the degree of user's preference (Pref) of a term t_i compared to the other terms' query by:

Pref (t_i) = [Number of close terms chosen for t_i in Q'/Number of terms of Q] + 1.

We add here the factor 1 to prevent those terms preferences which user didn't choose close terms are null. For the example above we have:

$$\begin{aligned}
&\mathrm{Pref}(t_1) = 3/3 + 1 = 2; \mathrm{Pref}\ (t_{11}) = 1; \mathrm{Pref}\ (t_{12}) = 1\\
&\mathrm{Pref}(t_{13}) = 1; \mathrm{Pref}\ (t_2) = 1; \mathrm{Pref}(t_3) = 2/3 + 1 = 5/3;\\
&\mathrm{Pref}(t_{31}) = 1; \mathrm{Pref}(t_{32}) = 1.
\end{aligned}$$

Here, it is clear that the term t_1 is more preferable than t_3 and t_2 because the user chooses the most significant number of close words of t_1. This is proves well that t_1 is the most important term in his research query. The term t_3 is also preferable for the user as the term t_2 which he didn't need close words for it.

Thus, the preferences calculated here are quite in conformity with the user' profile, because the most significant user's term is that which he seeks the maximum of close terms. In this manner, we can introduce the preferences between terms of the reformulated query in the relevance possibilistic of the documents of the collection.

We introduce these preferences between query' terms in our basic possibilistic model in the following way:

The quantitative relevance of each LE of a document (ELd_j) of the collection, knowing that the query is $Q' = (t_1, t_2, \ldots, t_T)$, is calculated in the following way: The formula (3) of section 4.7 becomes:

$$\begin{aligned}
\Pi'(\mathrm{ELd}_j|\mathrm{Q}') &= \Pi(t_1|\mathrm{ELd}_j) * \mathrm{Pref}(t_1) * \ldots * \Pi(t_T|\mathrm{ELd}_j) * \mathrm{Pref}(t_T)\\
&= \mathrm{nft}_{1j} * \mathrm{Pref}(t_1) * \ldots * \mathrm{nft}_{Tj} * \mathrm{Pref}(t_T) \qquad (3)
\end{aligned}$$

where $\mathrm{nft}_{ij} = \mathrm{tf}_{ij}/\max(\mathrm{tf}_{kj})$: the normalized frequency of the terms of the query in the LE.

The certainty to restore a LE of a relevant document $d_j(\mathrm{ELd}_j)$ for a query, noted $N(\mathrm{ELd}_j \mid Q')$, is given by the same manner as that presented in the section IV. G, except that the formula (8) becomes:

$$\Pi'(\neg\mathrm{ELd}_j|Q') = [(1 - \phi\mathrm{EL}_{1j})/\mathrm{Pref}(t_1)] * \ldots * [(1 - \phi\mathrm{EL}_{Tj})/\mathrm{Pref}(t_T)]. \qquad (8)$$

In fact, we introduced the factor Pref (t_i) into the calculation of the Possibility as well as the Necessity because this factor is quite related to the normalized frequency of the terms (nft_{ij}) in the required document.

Example

Let us consider a mini-collection of 3 documents d_1, d_2 and d_3:

$$d_1 = \{t_1, t_1, t_1, t_{11}, t_{11}, t_{12}, t_{12}, t_{12}, t_{13}, t_2, t_2, t_3, t_{31}\},$$
$$d_2 = \{t_1, t_1, t_1, t_{11}, t_{11}, t_{12}, t_{12}, t_{12}, t_{13}, t_2, t_2, t_3, t_{32}\},$$
$$d_3 = \{t_1, t_{11}, t_{11}, t_{12}, t_{12}, t_2, t_2, t_3, t_{31}, t_{32}, t_{32}\}.$$

These terms are distributed on the logical entities of these three documents, as Table IV indicates. User's profiles are given in Table V. The evaluation of the documents d_1, d_2 and d_3 for the query $Q'(t_1, t_{11}, t_{12}, t_{13}, t_2, t_3, t_{31}, t_{32})$ is given in Table VI.

Logical structure of document	d_1	d_2	d_3
ML	t_1, t_2	t_1, t_3	t_1, t_{32}
ML-1			t_2
ML-2	t_{12}		
ML-3		t_{12}	
ML-4			t_{31}
FL	t_{11}		
TL		t_{11}	
MSL	t_1, t_{12}	t_{13}, t_{12}	t_3, t_{32}
P	t_1, t_2, t_{13}, t_{31}, t_{12}, t_3, t_{11}	t_1, t_1, t_2, t_2, t_{32}, t_{11}, t_{12}	t_{11}, t_{11}, t_{12}, t_{12}, t_2

Table IV. Distribution of the terms in the logical structures of the three documents

Logical structure of document	User's profiles Coefficients α_j Profile1 (P1)	Coefficients α_j Profile2 (P2)	Coefficients α_j Profile3 (P3)
ML	10	2	2
ML-1	9	6	10
ML-2	8	5	4
ML-3	7	10	7
ML-4	6	4	9
FL	5	3	3
TL	4	9	6
MSL	3	7	5
P	2	8	8

Table V. The three user's profiles

	Profile1 (P1)			Profile2 (P2)			Profile3 (P3)		
	d_1	d_2	d_3	d_1	d_2	d_3	d_1	d_2	d_3
Without preferences between query's terms	14,66	13,46	14,55	16,38	20,7	12,31	14,74	16,66	15,17
Document's relevance order	1	3	2	2	1	3	3	1	2
With preferences between query's terms	17,8	17,3	18,02	18,44	26,38	16,14	16,4	22,34	18,78
Document's relevance order	2	3	1	2	1	3	3	1	2

Table VI. Results of the effect of the addition of preferences between query's terms

The SARIPOD System charges the entry agent to record the preferences between query's terms at the time of the interaction between user and system. In fact, these preferences enter well within the framework of the definition of its profile to the system. The results collected in Table VI show the importance of the definition of the preferences between query's terms for the case of the first profile (P1). Indeed, this factor was introduced like a multiplicative factor into the calculation of the possibility and like a quotient into calculation of the necessity; what makes it possible in consequence to increase the both scores of the possibility and the necessity at the same time.

Into case of not taken into account of preferences between terms and for the three profiles of Table V, the relevance's order of documents changes while passing from a profile to another. Whereas in the event of the taking into account of these preferences, only the first profile1 (P1) is significant and contributes to change relevance's order of documents. It's thanks to the term t_1 (having the preference 2 and existing in a LE having weight 10) and term t_2 (existing in a LE having weight 9) which contributed to increase the score of d_3 compared to others. For the both other profiles P2 and P3, the most preferable term (t_1) exists in a LE having weight 2; it's for this reason the factor Pref(t_1) didn't make the differences between documents' scores. This factor depends on coefficients of possibilistic relevance to define user's profile.

The insertion of the factors Pref(t_1) in calculations of the possibility and the necessity consists in increasing the scores of possibilistic relevance of the documents containing these terms with an aim of penalizing the scores of relevance of the documents not containing them. The penalization and the increase in the scores are proportional to the capacity of the terms to discriminate between the documents of the collection. In addition, these preferences make it possible to restore documents classified by preference of relevance. It is possible in this case to evaluate at which point a document d_1 is preferred than document d_2 or to measure the preference of a document d_1 compared to a whole of documents $\{d_3, d_4\}$. In fact, these factors are more effective than the factor idf, since the distribution of the terms in the document doesn't only depend on the presence or the absence of the terms in the documents of the collection (like idf), but of the distribution of their density in the documents of the collection. Thus, compared with idf these measurements are more powerful for negative discrimination.

7. Experimentation

SARIPOD system is integrated as a Java package in Jade multiagent platform. The classes' agents inherit their properties and their methods of the basic classes. In fact, we gathered all the useful functionalities of our SARIPOD system in only one convivial and interactive graphic interface (see Figure 2). The experimentation of an information processing system is the most significant stage for the improvement of its performance. Indeed, we chose to test our system through several axes to deduce some optimal parameter setting recommended for any user of our system. Indeed, we made the tests of SARIPOD system on some web sites and for keywords having a variable number of synonyms. Table VII gives our results. According to experimentations results, the determination of synonymic keywords is very dependent on the degree of cleaning of French dictionary *Le Grand Robert*, used as a data base of the lexical HSW. According to Table VII, we notice that any increase of the synonyms' number gives more chance to collect more URLs, and increases consequently the duration of the query treatment. Thanks to successive tests, we deduced that the optimal number of

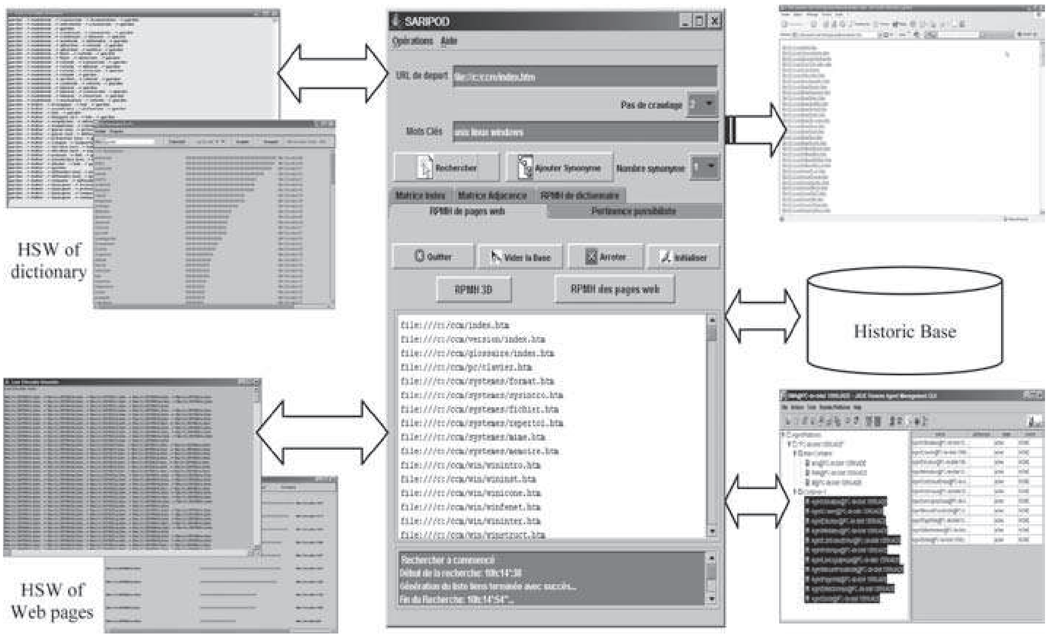

Figure 2.
Overview of SARIPOD system: input, output and control interfaces

Number of synonyms	Required keywords	Numbers of URLs obtained	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	15	39.22	7.33
1	*vérifier examiner*	93	17	29.74	5.77
2	*vérifier examiner voir*	207	20	16.23	3.69
3	*vérifier examiner voir éprouver*	363	21	9.53	2.87
4	*vérifier examiner voir éprouver reconnaître*	412	24	7.19	1.88
5	*vérifier examiner voir éprouver reconnaître essayer*	517	29	5.37	1.45
6	*vérifier examiner voir éprouver reconnaître essayer contrôler*	761	35	3.86	0.83
7	*vérifier examiner voir éprouver reconnaître essayer contrôler expérimenter*	833	42	1.34	0.67
8	*vérifier examiner voir éprouver reconnaître essayer contrôler expérimenter constater*	904	55	0.53	0.18

Table VII.
Experimentations results

words close to a keyword is: three, by taking account of the quality of the collected documents as well as the duration of the query treatment.

We also 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 Google web search engine answers to the query, by structuring it in a HSW so that, if a page among the returned answers seem relevant then all neighbors 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). According to Figure 3 and Table VIII, we clearly notice that the collected web pages in each search query are HSW. With equal density D, the L is small and C is high. In fact, the variations of L and C are not very important according to the numbers of collected web pages.

8. Synthesis and discussion

The use of the software agents for the search of information offers certain advantages compared to the current methods such as the search engines. Table IX recapitulates these advantages.

As evaluation of the SARIPOD system we present on one hand a justification of the use of the HSW graph in IR, and on the other hand a comparison with the traditional web search engine systems such as Google (Vise and Malseed, 2006). 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" 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

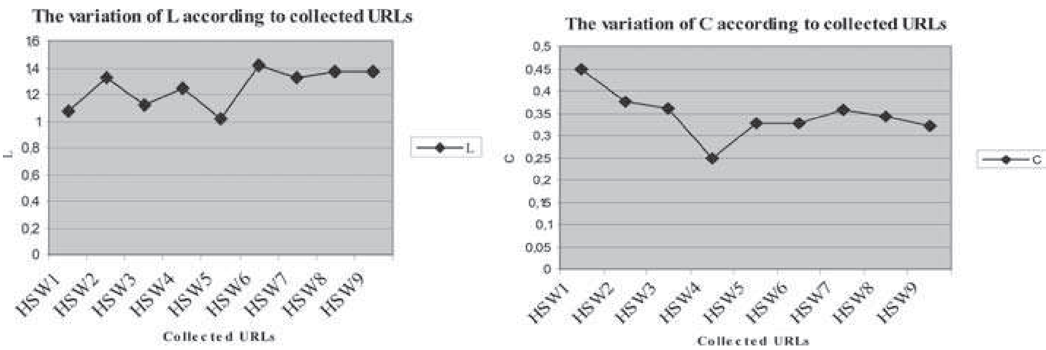

Figure 3.
The variations of L and C according to collected URLs

Table VIII.
The parameters L and C of the experiments HSW

HSW_i	The number of collected URLs	L	C
HSW_1	61	1.0690	0.4499
HSW_2	93	1.3305	0.3749
HSW_3	207	1.1167	0.3607
HSW_4	363	1.2497	0.2477
HSW_5	412	1.0148	0.3258
HSW_6	517	1.4124	0.3281
HSW_7	761	1.3215	0.3562
HSW_8	833	1.3685	0.3421
HSW_9	904	1.3724	0.3212

	The search engines	Software agents
Search criteria	The search of information is being based on one or more keywords. This supposes that the user is able to formulate his keywords exactly. In the contrary case, several non relevant information will be turned over and several relevant information will never be found	The agents are able to seek the information in a more intelligent way, for example seeking according to concepts. The agents are also able to correct user's query, basing on user's model or other information
Indexation	The indexing of information is made by collection of meta-information on information and the documents available on the web. It is an expensive method (in time and resources), ineffective and which does not correspond well to the dynamic nature of the internet	The agents can create their own knowledge bases which are updated after each research. If information changes site, the agents are able to find it and, thereafter, they adapt to this change. Moreover, the agents are able to communicate and cooperate between them (and it is their true force there), which accelerates and facilitates research
User interface	The search of information is often limited to some services (WWW). To find information offered by other services (data bases) often obliges the user to manage alone	The agents can remove the user from certain details, as the way with which a service must be handled. The user concentrates only on what he seeks, the agent deals with the remainder
Accessibility	The search engines are not always accessible, because of connection or congestion. The user will be then obliged to use one or more other search engines, so he will probably requires another way of proceeding	The agent resides on the user's machine, so it is always at the disposal of this one. An agent can carry out several tasks in the day and at night, and sometimes it will be able to carry out them in parallel. The advantage of such an agent also lies in the fact that it is intelligent (because it takes account of a dynamic profile of the user) and that it can consequently try to avoid the peak hours
Adaptability	Information on the network is very dynamic; often the search engines refer to information whose locality changed. The search engines do not learn and do not adapt to the users. Moreover, the user cannot receive the updates of information. To make search for information in such a way is very expensive	The agents adapt to the preferences and the wishes of each user. They can thus learn from their preceding research and thereafter understand better the users' needs

Table IX.
Comparative study between search engines and Software agents

answer this same query. The essential difference is that each cloud of web pages strongly answers the query in a particular way. We can compare our results with the following lexical phenomenon: the query "vérifier", in the HSW of french synonyms, gives four clouds of verbs close to "vérifier": the first cloud concerns $A = \{$examiner, voir, éprouver, reconnaître, $\dots\}$, the second $B = \{$essayer, contrôler, expérimenter, s'assurer, $\dots\}$, etc. for the two others. In an analogical way, it's the same thing for the web: 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 "vérifier" if A is interested in the "examen", as B which also answers pertinently the same query "vérifier" if B is interested in the "contrôle", 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 single 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 sub-HSW. With the deepest of this structure we find single web pages. The set of answers was thus organized in HSW and sub-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 related thus include the synonyms of this word but don't narrow down to them. There will be potentially all the words more or less close to query's keyword (it will be actually limited by an acceptable lowest proximity threshold number). 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 does not 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 *et 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). Our system SARIPOD takes again this definition and calculates 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.

9. Conclusion

This paper proposes a new qualitative approach for an intelligent possibilistic web IR system using multiagent approach based on HSW and PN. The first HSW consists in:

- Given a query, we carry out a classification of the answers web pages (we thus create a kind of domain and sub-domain).
- On the same set of web pages, another query (without relationship with the preceding one) would have led to another classification.
- We can wonder legitimately if there are as many classifications as of queries.
- We can wonder about the parties taken of the various engines (Google, Yahoo, Voilà, Alta-Vista, . . .) on classifications which they pose a priori.

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.

Indeed, SARIPOD is a new approach for IR Model based on possibility theory. This model encodes relationship dependencies existing between query terms and web documents through naïve PN and quantifies these relationships by two measures: possibility and necessity. The possibility degree is convenient to filter documents out from the response and the necessity degree is useful for document relevance confirmation. The retrieved documents are those which are necessarily or possibly relevant given a user's query. The search process restores the plausibly or necessarily relevant documents for a user need. The user's query is seen as new information to propagate in a possibilistic network. Moreover, if the basic approach takes account of the quantitative aspect, our system extends it to the qualitative possibilistic framework, by introducing preferences between query terms.

Our tests are only limited to the verbal occurrences in the dictionary HSW of verbs definitions, but we plan to extend the tests to other grammatical categories, like refining our approach for the substantives by considering for example also the verbal occurrences in names definitions, etc. We also plan to carry out finer measurements of the performances of SARIPOD system by extending the tests with other types of web documents.

In the integration of dependence relations between documents' terms, the edges are measured by numerical values translating the quantities and not partial orders. In order to quantify these relations, we could base ourselves on the knowledge represented in ontology. In fact, ontology makes it possible to formalize semantic links between concepts (Ben Ahmed, 2007). Defined within a possibilistic framework, it could add relevant information to consider during the process of propagation started by the query. The network would be composed of sub-network of documents and of a sub-network of query terms. These sub-networks could be connected through ontology.

References

Awada, A. and Chebaro, B. (2004), "Etude de la synonymie par l'extraction de composantes N-connexes dans les graphes de dictionnaires", paper presented at JEL2004, Nantes.

Awada, A. (2005), "Regroupement de synonymes en composantes de sens dans un dictionnaire", *Dans le Proceedings of IC2005, France*.

Ben Ahmed, M. (2007), *Cognition entre Philosophie, Science et Technologie*, Centre de Publication Universitaire (CPU), Tunis.

Benferhat, S., Dubois, D., Garcia, L. and Prade, H. (2002), "On the transformation between possibilistic logic bases and possibilistic causal networks", *International Journal of Approximate Reasoning*, Vol. 29 No. 2, pp. 135-73.

Berry, M.J.A. and Linof, G. (1997), *Data Mining: Techniques Appliquées au Marketing, à la vente, et aux Services Clients*, InterEditions, Paris.

Borgelt, C., Gebhardt, J. and Kruse, R. (2000), "Possibilistic graphical models", *Computational Intelligence in Data Mining, CISM Courses and Lectures 408*, pp. 51-68.

Brini A.H., Boughanem M. and Dubois D. (2005), "A model for information retrieval based on possibilistic networks", *Proceedings of the SPIRE 2005, Buenos Aires (Argentine)*, LNCS, Springer-Verlag, New York, NY, pp. 271-82.

Brini, A.H. and Boughanem, M. (2003), "Relevance feedback: introduction of partial assessments for query expansion", *Proceedings of the 2nd EUSFLAT Conference*, *Zittau*, pp. 67-72.

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

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

Dubois, D. and Prade, H. (1998), "Possibility theory: qualitative and quantitative aspects", in Gabbay, D.M. and Smets, P. (Eds), *Dans: Quantified Representation of Uncertainty and Imprecision, Handbook of Defeasible Reasoning and Uncertainty Management Systems*, Vol. 1, Kluwer, Dordrecht, pp. 169-226.

Ferber, J. (1995), *Les systèmes Multi-Agents, Vers Une Intelligence Collective*, InterEditions, Paris.

Gaume, B., Hathout, N. and Muller, P. (2004), "Désambiguïsation par proximité structurelle", paper presented at TALN 2004, Fès.

Miller, R.C. and Bharat, K. (1998), "SPHINX: a framework for creating personal, site-specific web crawlers", *Proceedings of the 7th International World Wide Web Conference (WWW7)*, *Brisbane*, *Australia*, *1998*, printed in *Computer Network and ISDN System*, Vol. 30, pp. 119-30.

Newman, M.E.J. (2003), "The structure and function of complex networks", *SIAM Review*, Vol. 45, pp. 167-256.

Ribeiro-Neto, B., Silva, I. and Muntz, R. (1996), "A belief network model for IR", *Proceedings of the 19th ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 253-60.

Salton, G., Allan, J., Buckley, C. and Singhal, A. (1994), "Automatic analysis, theme generation and summarization of machine readable texts", *Science*, Vol. 264 No. 3, pp. 1421-26.

Sergi, V. and Ricard, V.S. (2007), "Hierarchical small-worlds in software architecture", *Dynamics of Continuous Discrete and Impulsive Systems: Series B; Applications and Algorithms*, Vol. 14, pp. 1-11.

Singhal, A., Salton, G., Mitra, M. and Buckley, C. (1996), "Document length normalization", *Information Processing and Management*, Vol. 32 No. 5, pp. 619-33.

Sparck, J.K. (1998), "A look back and a look forward", *Proceedings of the 11th Annual international ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 13-29.

Vise, D. and Malseed, M. (2006), *Google Story*, Dunod, Paris.

Watts, D. and Strogatz, S. (1998), "Collective dynamics of 'small-world' networks", *Nature*, Vol. 393, pp. 440-2.

Zadeh, L.A. (1978), "Fuzzy sets as a basis for a theory of possibility", *Fuzzy Sets and Systems*, Vol. 1, pp. 3-28.

Zaghdoud, M. (2003), "SAFIINA: Système basé multi-Agent de Fusion d'Informations INcertAines", thèse de Doctorat en Informatique, Ecole Nationale des Sciences de l'Informatique, Tunis.

Further reading

Matthew, O.J. and Brian, W.R. (2004), "Search and the strategic formation of large networks: when and why do we see power laws and small worlds", Lee Center for Advanced Networking, Pasedena, CA.

Nguyen, V. and Martel, C. (2003), *Analysing and Characterizing Small-worlds Graphs*, NFS grant CCR-85961, Computer Science, UC Davis, CA.

Prade, H. and Testemale, C. (1987), "Application of possibility and necessity measures to documentary information retrieval", in Bouchon, B. and Yager, R. (Eds), *Uncertainty in Knowledge-Based Systems*, LNCS No. 286, Springer Verlag, Berlin, p. 265-74.

About the authors

Bilel Elayeb has been a PhD Student at Institut National Polytechnique de Toulouse (INPT), France and has been at Ecole Nationale des Sciences de l'Informatique (ENSI) of La Manouba, Tunisia since 2005. He obtained a Master thesis in Computer Science from Ecole Nationale des Sciences de l'Informatique of La Manouba in 2004. His research focuses on information retrieval and multiagent system, including possibility theory and Hierarchical Small-Worlds graphs. He has been a member of the Logic, Interaction, Language, and Computation (LILaC) research group at the Institut de Recherche en Informatique de Toulouse (IRIT), France since 2005 and of the RIADI-GDL research laboratory since 2002. He has taught computer science at Manouba University in Tunisia since 2002. Bilel Elayeb is the corresponding author and can be contacted at: bilel.elayeb@riadi.rnu.tn

Fabrice Evrard has been an Assistant Professor at ENSEEIHT, Toulouse, France since 1983. His research focuses on multiagent system, dictionary, information retrieval and Hierarchical Small-Worlds graphs. He supervised a master degree in artificial intelligence at Institut National Polytechnique de Toulouse (INPT). He conducted Le Groupe Raisonnement, Action et Actes de Langage (GRAAL) team working which is a part of LILaC research group at the Institut de Recherche en Informatique de Toulouse (IRIT), France. He has supervised many masters thesis and PhD thesis in artificial intelligence and multiagent system.

Montaceur Zaghdoud has been an Assistant Professor at Ecole Nationale des Sciences de l'Informatique (ENSI) of La Manouba, Tunisia since 2004. He obtained a PhD thesis in computer science from ENSI in 2003. His research focuses on artificial intelligence, uncertain theory, information retrieval and multiagent system. He has been member of the RIADI-GDL research laboratory since 1992. He has supervised many masters thesis and PhD thesis in artificial intelligence and multiagent system. He has been the headmaster of the Institut Supérieur en Informatique (ISI) of Kef in Tunisia since 2007.

Mohamed Ben Ahmed is Emeritus Professor at Ecole Nationale des Sciences de l'Informatique (ENSI) of La Manouba, Tunisia. He has been the headmaster of the RIADI-GDL research laboratory in ENSI, Tunisia since 1992. His research focuses on artificial intelligence, uncertain theory, documentation, information retrieval, linguistic and multiagent system. He has directed many masters thesis and PhD thesis in these domains. He has conducted many international research projects in the domain of e-learning, e-commerce and documentation with cooperation between ENSI in Tunisia and INRIA, IRIT and CNRS in France.