International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems
Vol. 9, Suppl. (Sep. 2001) 77–87

AN INFORMATION RETRIEVAL MODEL WITH ORDINAL LINGUISTIC WEIGHTED QUERIES BASED ON TWO WEIGHTING ELEMENTS

E. HERRERA-VIEDMA
Department of Computer Sciences and A.I.
University of Granada, 18071 – Granada, Spain
E-mail: viedma@decsai.ugr.es

Received February 2001
Revised June 2001

An Information Retrieval (IR) model defined using an ordinal fuzzy linguistic approach is proposed. It accepts ordinal linguistic weighted queries based on two weighting elements: the query terms and the query sub-expressions. In such a way, users may easily express simultaneously several semantic restrictions in a query. A symmetrical threshold semantic is associated to the weights of the query terms and an importance semantic is associated to the weights of the query sub-expressions. The advantage of this IR model with respect to others is the facility for expressing different semantic restrictions on the desired documents simultaneously, incorporating more flexibility in the user-IR system interaction.

Keywords: Linguistic modelling, fuzzy information retrieval, linguistic querying.

1. Introduction

The main activity of an IR system (IRS) is the gathering of pertinent archived documents that best satisfy the user queries. IRSs present three components to carry out their activity:

1.- *A Database:* which stores the documents and the representation of their information contents (index terms).

2.- *A Query Subsystem:* which allows users to formulate their queries by means of a query language.

3.- *An Evaluation Subsystem:* which evaluates the documents for a user query obtaining a Retrieval Status Value (RSV) for each document.

The query subsystem supports the user-IRS interaction, and therefore, it should be able to account for the imprecision and vagueness typical of human communication. This aspect may be modelled by means of the introduction of weights in the query language. Many authors have proposed weighted IRS models using Fuzzy Set Theory [2,3,5,6,7,8,11,13,15]. Usually, they assume numeric weights (values in [0,1]). However, the use of query languages based on numeric weights forces the user to quantify qualitative concepts (such as "importance"), ignoring that many users are not able to provide their information needs precisely in a quantitative form but in a qualitative one. In fact, it seems more natural to characterize the contents of desired documents by explicitly associating a linguistic descriptor to a term in a query, like "important" or "very important", instead of a numerical value. In this sense, some fuzzy linguistic IRS models [4,12] have been proposed using a *fuzzy linguistic approach* [16] to model the query weights and document scores. A useful fuzzy linguistic approach which allows us to reduce the complexity of the design for the IRSs [10] is called the *ordinal fuzzy linguistic approach* [9]. In this approach, the query weights and document scores are ordered linguistic terms.

On the other hand, we have to establish the semantic associated to the query weights to formalize fuzzy linguistic weighted querying. There are four semantic possibilities [3,10,12]: i) weights as a measure of the importance of a specific element in representing the query, ii) as a threshold to aid in matching a specific document to the query, iii) as a description of an ideal or perfect document, and iv) as a limit on the amount of documents to be retrieved for a specific element. Usually, in a weighted query most query subsystems use only one of the above semantic possibilities. However, users may need to express different kinds of semantic restrictions in a weighted query. In [10] we present a fuzzy linguistic IRS that accepts weighted queries based only on one weighting element (query terms) which has simultaneously several weighting semantics associated to it. The difficulty of this model is that the simultaneous use of many semantics on a same element of query is not easy for non-expert users, and furthermore, its design is complicated.

In this paper, we present a fuzzy linguistic IR model that overcomes the difficulties observed in [10]. Its query language allows users to weigh two elements of a query simultaneously: the query terms and the query sub-expressions (associations of terms related by Boolean logical connectives AND ($\wedge$), OR ($\vee$), and NOT ($\neg$). Users can express their requests by means of ordinal linguistic weighted queries based only on two semantics: a *symmetrical threshold semantic* associated to the term weights, and an *importance semantic* associated to the sub-expression weights. The first one is modelled by a linguistic matching function that is easier than the one we proposed in [10]. The latter is modelled by means of two aggregation operators of ordinal weighted linguistic information, *the Linguistic Weighted Disjunction operator* and the *Linguistic Weighted Conjunction operator* [9].

The paper is set out as follows. The ordinal fuzzy linguistic approach is presented in Section 2. The fuzzy linguistic IR model is defined in Section 3. Finally, Section 4 draws our conclusions.

2. The Ordinal Fuzzy Linguistic Approach

The *ordinal fuzzy linguistic approach* is an approximate technique appropriate to deal with qualitative aspects of problems, which is based on the concept of fuzzy linguistic variable [16]. An ordinal fuzzy linguistic approach is defined by considering a finite and totally ordered label set $S = \{s_i\}, i \in \{0, \ldots, \mathcal{T}\}$ in the usual sense ($s_i \geq s_j$ if $i \geq j$) and with odd cardinality (7 or 9 labels) as in [1]. The mid term representing an assessment of "approximately 0.5" and the rest of the terms being placed symmetrically around it [1]. The semantic of the linguistic term set is established from the ordered structure of the term set by considering that each linguistic term for the pair $(s_i, s_{\mathcal{T}-i})$ is equally informative. The semantic of the labels is given by fuzzy numbers defined on the [0,1] interval, which are described by linear trapezoidal membership functions represented by the 4-tuple $(a_i, b_i, \alpha_i, \beta_i)$ (the first two parameters indicate the interval in which the membership value is 1.0; the third and fourth parameters indicate the left and right widths of the distribution). Furthermore, we require the following operators:

1. $Neg(s_i) = s_j, \; j = \mathcal{T} - i$.
2. $MAX(s_i, s_j) = s_i$ if $s_i \geq s_j$.
3. $MIN(s_i, s_j) = s_i$ if $s_i \leq s_j$.

For example, we can use the following set of nine labels with its associated semantic to evaluate the linguistic variables in the fuzzy linguistic IR model [1] (see Figure 1):

$$
\begin{aligned}
&T = Total = (1, 1, 0, 0)\\
&EH = Extremely\_High = (.98, .99, .05, .01)\\
&VH = Very\_High = (.78, .92, .06, .05)\\
&H = High = (.63, .80, .05, .06)\\
&M = Medium = (.41, .58, .09, .07)\\
&L = Low = (.22, .36, .05, .06)\\
&VL = Very\_Low = (.1, .18, .06, .05)\\
&EL = Extremely\_Low = (.01, .02, .01, .05)\\
&N = None = (0, 0, 0, 0).
\end{aligned}
$$

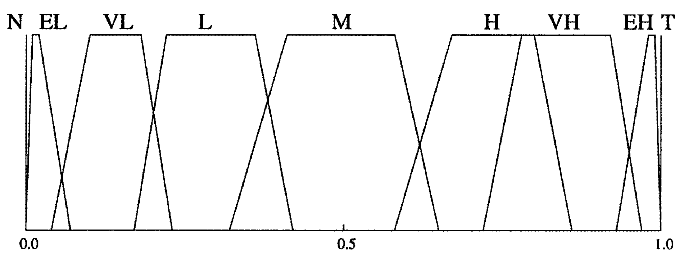

Figure 1: A Set of Nine Terms with Its Semantic

On the other hand, in this contribution we have to deal with ordinal linguistic weighted information in the evaluation of user queries, and thus, we need some aggregation operators of ordinal linguistic weighted information to obtain the RSVs. We propose to use the following ones [9]:

Definition 1.- *A set of ordinal linguistic weighted opinions, $\{(c_1, a_1), \ldots, (c_m, a_m)\}$, is aggregated by the Linguistic Weighted Disjunction (LWD) according to this expression:*

$$LWD[(c_1, a_1), \ldots, (c_m, a_m)] = MAX_i \ \{MIN(c_i, a_i)\},$$

where $c_i, a_i \in S$, a_i represents the weighted opinion and c_i the importance degree of a_i.

Definition 2.- *A set of ordinal linguistic weighted opinions, $\{(c_1, a_1), \ldots, (c_m, a_m)\}$, is aggregated by means of the Linguistic Weighted Conjunction (LWC) operator according to this expression:*

$$LWC[(c_1, a_1), \ldots, (c_m, a_m)] = MIN_i \ \{MAX(Neg(c_i), a_i)\}.$$

3. The Fuzzy Linguistic IR Model

In this section, we present a fuzzy linguistic IR model that supports ordinal linguistic weighted queries based on two weighting elements. In the following subsections, we present their constituents.

3.1. *Definition of Database*

We assume a database of a traditional fuzzy IRS as in [7,13,15], where the IRS-user interaction is unnecessary because it is built automatically. Therefore, we do not use an ordinal fuzzy linguistic formulation for the database.

The database stores the finite set of documents D= $\{d_1, \ldots, d_m\}$, their respective representations R(D)= $\{R_{d_1}, \ldots, R_{d_m}\}$, and the finite set of index terms T= $\{t_1, \ldots, t_l\}$. Documents are represented by means of index terms, which describe the subject content of the documents. The representation of a document is a fuzzy set of terms characterized by a numeric indexing function $F : \text{DxT} \rightarrow [0, 1]$, which is called *index term weight* [15]. F maps a given document d_j and a given index term t_i to a numeric weight between 0 and 1. $F(d_j, t_i) = 0$ implies that the document d_j is not at all about the concept(s) represented by index term t_i and $F(d_j, t_i) = 1$ implies that the document d_j is perfectly represented by the concept(s) indicated by t_i. Using the numeric values in (0,1) F can weigh index terms according to their significance in describing the content of a document in order to improve the document retrieval. Thus $F(d_j, t_i)$ is a numerical weight that represents the degree of significance of t_i in d_j. The quality of the retrieval results strongly depends on the criteria used to compute F. Different document term weighting schemes have been used for defining F [3,14]. In this paper, we do not focus this aspect and assume any of existing weighting methods.

3.2. *Definition of Query Subsystem*

As was aforementioned, the terms of a query may be weighted according to four semantic possibilities [3,10,12]: threshold semantic, importance semantic, perfection semantic and quantitative semantic. Usually, weighted query subsystems accept in a query either one semantic or another, but not all at the same time. However, a user may need to express simultaneously different semantic restrictions in a query. In [10] we propose a weighted query subsystem that accepts to weigh the query terms using several semantics simultaneously. The use of this model is not easy, and sometimes, non-expert users may have confusion problems when they want to apply the different semantics associated to a same element of query at the same time. On the other hand, we can observe that a Boolean query presents four elements that a user may weigh to specify restrictions on the documents [10,12]: terms, sub-expressions, connectives and the whole query. However, most weighted query subsystems proposed in the literature [2,3,10,12,15] accept mainly to weigh the first element, i.e., the terms of query.

In order to overcome the difficulties of fuzzy linguistic IRS presented in [10], we propose a weighted query subsystem that supports ordinal linguistic weighted queries based simultaneously on two of the above weighting elements, terms and sub-expressions. Each user query is expressed as a combination of the index terms which are connected by the logical operators AND ($\wedge$), OR ($\vee$), and NOT ($\neg$), being the query terms and query sub-expressions weighted by means of ordinal linguistic values. The weights of query terms and the weights of the query sub-expressions are associated to the following semantics, respectively:

1. *Symmetrical threshold semantic* [10]. This semantic defines query weights as requirements of satisfaction of each term of query to be considered in matching document representations to the query. By associating threshold weights to terms in a query, the user is asking to see all documents sufficiently about the topics represented by such terms. In practice, he requires to reward a document whose index term weights F exceed the established thresholds with a high RSV, but allowing some small partial credit for a document whose F values are lower than the thresholds. Then, the query weights indicate presence requirements, i.e., they are presence weights. *Symmetrical threshold semantic* is a special threshold semantic which assumes that a user may use presence weights or absence weights in the formulation of weighted queries. Then, it is symmetrical with respect to the mid threshold value, i.e., it presents the usual behaviour for the threshold values which are on the right of the mid threshold value (presence weights), and the opposite behaviour for the values which are on the left (absence weights or presence weights with low value).

2. *Importance semantic* [2,15]. This semantic defines sub-expression weights as a measure of the relative importance of each sub-expression of query with respect to the others in the query. By associating relative importance weights to sub-expressions in a query, the user is asking to see all documents whose

content represents more the concept associated to the most important sub-expressions than to the less important ones. In practice, this means that the user requires that the computation of the RSV of a document is dominated by the more heavily weighted sub-expressions.

3.2.1. *Rules for Formulating Ordinal Linguistic Weighted Queries*

Formally, in [4] a fuzzy linguistic weighted Boolean query with one single semantic for the weights was defined as any legitimate Boolean expression whose atomic components are pairs $< t_i, c_i >$; t_i is an element of the set T of terms, and c_i is a value of the linguistic variable, *Importance*, qualifying the importance that term t_i must have in the desired documents.

Similarly, we use the linguistic variable *Importance* to model linguistic weights of queries. However, we define the linguistic variable by an ordinal fuzzy linguistic approach, as was shown in Section 2. Thus, we assume a set of ordinal linguistic terms S to express the linguistic weights. So, we use the same linguistic variable to model both weighting semantics, but with a different interpretation. For example, a query term t_i with threshold weight of value "*High*" means that user requires documents in whose content t_i should have at least a high importance value; however, a sub-expression with importance weight of value "*High*" means that user requires that in the computation of the set of retrieved documents the meaning of sub-expression must have a high importance value.

In order to overcome the difficulties of equivalence that may appear in the weighted Boolean queries [15,8], the user queries may only be expressed either in a conjunctive normal form (CNF) or in a disjunctive normal form (DNF). Accordingly, the set Q of the legitimate queries is defined by the following syntactic rules:

1. $\forall q^1 =< t_i | \neg t_i, c_i > \rightarrow q^1 \in$ Q, where $c_i \in S$ is the threshold ordinal linguistic weight assigned by a user to the term t_i. This rule defines simple queries.

2. $\forall q^2 = \wedge_{k=1}^{n \geq 2} q_k^1, \rightarrow q^2 \in$ Q. This rule defines the queries expressed by conjunctive sub-expressions.

3. $\forall q^3 = \vee_{k=1}^{n \geq 2} q_k^1, \rightarrow q^3 \in$ Q. This rule defines the queries expressed by disjunctive sub-expressions.

4. $\forall q^4 = \vee_{p=1}^{m \geq 2} (q_p^2 | q_p^1, c_p), \rightarrow q^4 \in$ Q, where $c_p \in S$ is the importance ordinal linguistic weight assigned by a user to the conjunctive sub-expression q_p^2 or to the atom q_p^1. This rule defines the complex queries expressed by DNFs.

5. $\forall q^5 = \wedge_{p=1}^{m \geq 2} (q_p^3 | q_p^1, c_p), \rightarrow q^5 \in$ Q. This rule defines the complex queries expressed by CNFs.

6. All legitimate queries are only those obtained by applying rules 1-5 only.

As in [8,10], we assume that a term may appear several times in the same query, and therefore, the query subsystem accepts the possibility of having queries with different weights on the same terms.

3.3. *Definition of Evaluation Subsystem*

The evaluation subsystems for weighted Boolean queries with more than one term act by means of a constructive bottom-up process based on the *criterion of separability* [8,15]. The problems appear when the evaluation of an atom depends on the evaluations of other atoms of query, because then the separability property is not satisfied by the evaluation mechanism, and in such a case, a bottom-up process may not be applied. For example, this is the situation in fuzzy IRSs where weights are associated to an importance semantic [2,7]. In our fuzzy linguistic IR model this difficulty does not appear. We apply the importance semantic in the query sub-expressions and they are evaluated jointly in the last step of evaluation using the adequate weighted aggregation operators defined in Section 2.

Therefore, we propose an constructive bottom-up evaluation subsystem that evaluates documents in terms of their relevance to an ordinal weighted linguistic query based on two weighting elements. It satisfies the separability property at the same time as supporting the semantics of the two weighting elements. It acts in two steps:

- Firstly, the documents are evaluated according to their relevance only to atoms of the query. In this step, a partial RSV is assigned to each document with respect to each atom in the query.

- Secondly, the documents are evaluated according to their relevance to Boolean combinations of atomic components (their partial RSVs), and so on, working in a bottom-up fashion until the whole query is processed. In this step, a total RSV is assigned to each document with respect to the whole query.

This evaluation subsystem presents the following characteristics:

1. The RSVs are ordinal linguistic values taken from the linguistic variable *Importance* but, in this case, representing the concept of *relevance.* Therefore, the set of linguistic terms S is also assumed to represent the relevance values.

2. The terms appearing in the queries are considered only, and thus the absent terms are not considered in the evaluation.

3. The symmetrical threshold semantic is applied in the evaluation of atoms. According to this semantic the evaluation subsystem assumes that a user may search for documents with a minimally acceptable presence of one term in their representations (as in [12]) or documents with a maximally acceptable presence of one term in their representations. Then, when a user asks for documents in which the concept(s) represented by a term t_i is (are) with the value *High Importance*, the user would not reject a document with a F value greater than *High*; on the contrary, when a user asks for documents in which

the concept(s) represented by a term t_i is (are) with the value *Low Importance*, the user would not reject a document with a F value less than *Low*. Given a request $< t_i, c_i > \in$ TxS, this means that the query weights that imply the presence of a term in a document $c_i \geq s_{T/2}$ (e.g. *High, Very High,*) they must be treated differently to the query weights that imply the absence of one term in a document $c_i < s_{T/2}$ (e.g. *Low, Very Low*). Then, if $c_i \geq s_{T/2}$ the request $< t_i, c_i >$, is synonymous with the request $< t_i, \textit{at least } c_i >$, which expresses the fact that the desired documents are those having F values as high as possible; and if $c_i < s_{T/2}$ is synonymous with the request $< t_i, \textit{at most } c_i >$, which expresses the fact that the desired documents are those having F values as low as possible. This interpretation was defined in [10] by means of a complicated parameterized linguistic matching function $g : D\text{x } \text{Tx}S \to S$. In this contribution, we redefine that function by means of an easier expression. Then, given an atom $< t_i, c_i > \in$ TxS and a document $d_j \in D$, g establishes how well the index term weight $F(d_j, t_i)$ of document d_j satisfies the request expressed by the linguistic weight c_i of atom $< t_i, c_i >$ according to the following expression:

$$g(d_j, < t_i, c_i >) = \begin{cases} s_{Min\{a+\mathcal{B},T\}} & \text{if } s_{T/2} \leq s_b \leq s_a \\ s_{Max\{0,a-\mathcal{B}\}} & \text{if } s_{T/2} \leq s_b \text{ and } s_a < s_b \\ Neg(s_{Max\{0,a-\mathcal{B}\}}) & \text{if } s_a \leq s_b < s_{T/2} \\ Neg(s_{Min\{a+\mathcal{B},T\}}) & \text{if } s_b < s_{T/2} \text{ and } s_b < s_a \end{cases}$$

such that, (i) $s_b = c_i$; (ii) s_a is the linguistic index term weight obtained as $s_a = Label(F(d_j, t_i))$, being $Label : [0,1] \to S$ a function that assigns a label in S to a numeric value $r \in [0,1]$ according to the following expression:

$$Label(r) = Sup_q\{s_q \in S : \ \mu_{s_q}(r) = Sup_v \ \{\mu_{s_v}(r)\}\};$$

and (iii) $\mathcal{B}$ is a bonus value that rewards/penalizes the partial RSV of d_j for the satisfaction/dissatisfaction of request $< t_i, c_i >$, which can be defined in an independent way, for example as $\mathcal{B} = 1$, or depending on the closeness between $Label(F(d_j, t_i))$ and c_i, for example as $\mathcal{B} = round(\frac{2(|b-a|)}{T})$. We must point out that g is different from the usual matching functions for threshold semantics proposed in the literature, which are monotone non-decreasing functions. g is symmetrical with respect to the mid threshold value ($s_{T/2}$). That is, g is non-decreasing in $Label(F(d_j, t_i))$ for the threshold values which are on the right of $s_{T/2}$, and monotone decreasing in $Label(F(d_j, t_i))$ for the threshold values which are on the left of $s_{T/2}$.

4. As queries are put into CNF or DNF form, only atoms in a query are negated. When we have an atom with a negated index term we can negate the weighted term or weigh the negated term. As was done in [7,10], the NOT operator is modelled according to the latter interpretation. This means that the evaluation of document d_j for a negated atom $< \neg(t_i), c_i >$ is obtained from the negation of the index term weight, i.e., $1 - F(d_j, t_i)$.

5. The importance semantic in a query formed by one sub-expression has no meaning because this semantic defines the query weights as measures of the "relative importance" of each sub-expression with respect to the others in the query.

6. The evaluation subsystem distinguishes two kinds of logical connectives: i) the non-weighted logical connectives, which establish simple relations among the atoms into subexpressions of a query, and ii) the weighted logical connectives, which establish importance relations among the sub-expressions of a query (see Figure 2).

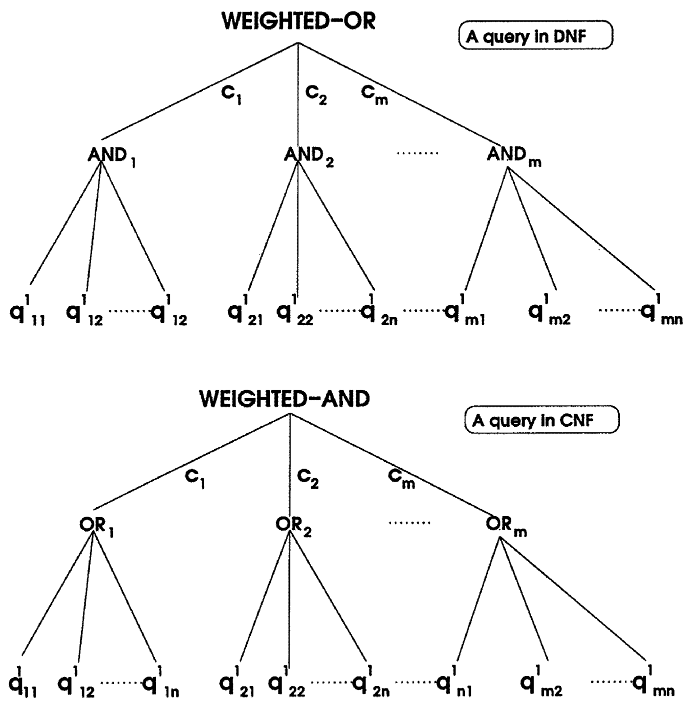

Figure 2: Queries in Normal Form.

7. The importance semantic associated to the query sub-expressions is modelled when weighted logical connectives AND ($\wedge$) and OR ($\vee$) are applied in the evaluation of query. As was aforementioned, these logical connectives are modelled by means of the aggregation operators of ordinal linguistic weighted information LWC and LWD, respectively. These operators incorporate the relative importance semantic in their actions guaranteeing its correct application. They act in such a way that the more important the query sub-expressions, the more influential they are in determining the final result. To do so, they apply the importance semantic restrictions by means of their

transformation functions [9]. On the other hand, according to their definitions, when a user does not want to impose importance restrictions on the connected sub-expressions, he must assign the linguistic importance weights s_0 to the conjunctions and $s_{\mathcal{T}}$ to the disjuncts. We should note that these operators overcome some limitations of classical evaluation mechanisms defined to deal with the importance semantic, e.g., the problems of the AND connective when it is modelled using the fuzzy connective MIN [2].

8. Finally, the non-weighted logical connectives AND and OR are modelled by means of the linguistic functions MIN and MAX, respectively.

Assuming the above properties, the evaluation subsystem can be synthesized by means of a linguistic evaluation function E : D x Q $\rightarrow S$, which evaluates the different kind of queries, q^k, $k \in \{1,2,3,4,5\}$, according to the following five rules:

1. $E(d_j, q^1) = g(d_j, q^1)$.
2. $E(d_j, q^2) = MIN(E(d_j, q_1^1), \ldots, E(d_j, q_n^1))$.
3. $E(d_j, q^3) = MAX(E(d_j, q_1^1), \ldots, E(d_j, q_n^1))$.
4. $E(d_j, q^4) = LWD[(c_1, E(d_j, q_1^h)), \ldots, (c_m, E(d_j, q_m^h)], \; h \in \{1,2\}$.
5. $E(d_j, q^5) = LWC[(c_1, E(d_j, q_1^h)), \ldots, (c_m, E(d_j, q_m^h)], \; h \in \{1,3\}$.

Then, the issue of system for any user query is a fuzzy subset of documents characterized by the linguistic membership function E :

$$\{(d_1, E(d_1, q^k)), \ldots, (d_m, E(d_m, q^k))\}.$$

The documents are shown in decreasing order of E and arranged in linguistic relevance classes, in such a way that the maximal number of classes is limited by the cardinality of the set of labels $(\mathcal{T}+1)$ chosen for the linguistic variable *Importance*.

4. Conclusions

In this paper, we have presented a fuzzy linguistic IR model based on an ordinal fuzzy linguistic approach that overcomes the difficulties observed in [10] and improves the user-IRS interaction.

So, users may represent their information needs by means of ordinal linguistic weighted queries based on two weighting elements: the terms of query and the sub-expressions of query. Then, they may easily express simultaneously several semantic restrictions on their desired documents in a query. A symmetrical threshold semantic is associated to the query terms and an importance semantic is associated to the query sub-expressions. In such a way, the expression potential of query language is increased. To manage this kind of weighted queries, we have designed an evaluation subsystem that integrates coherently the linguistic weights with respect to both semantics. This subsystem is based on the classical bottom-up evaluation mechanism. The IRS presents the retrieval issues by means of a limited number of ordinal linguistic relevance classes.

In the future, we shall study the use of the rest of weighting elements that a Boolean query presents, as for example the connectives and the whole query.

Acknowledgements

The author acknowledge partial support of the Spanish CYCIT project PB-981305. He is also grateful to the anonymous referees for their comments.

References

1. P.P. Bonissone and K.S. Decker, Selecting uncertainty calculi and granularity: An experiment in trading-off precision and complexity, in: L.H. Kanal and J.F. Lemmer, Eds., *Uncertainty in Artificial Intelligence* (North-Holland, 1986) 217-247.
2. A. Bookstein, Fuzzy request: An approach to weighted Boolean searches, *Journal of the American Society for Information Science* **31** (1980) 240-247.
3. G. Bordogna, C. Carrara and G. Pasi, Query term weights as constraints in fuzzy information retrieval, *Information Processing & Management* **27** (1991) 15-26.
4. G. Bordogna and G. Pasi, A fuzzy linguistic approach generalizing Boolean information retrieval: A model and its evaluation, *Journal of the American Society for Information Science* **44** (1993) 70-82.
5. G. Bordogna and G. Pasi, Linguistic aggregation operators of selection criteria in fuzzy information retrieval, *International Journal of Intelligent Systems* **10** (1995) 233-248.
6. D. Buell and D.H. Kraft, Threshold values and boolean retrieval systems, *Information Processing & Management* **17** (1981) 127-136.
7. D. Buell and D.H. Kraft, A model for a weighted retrieval system, *Journal of the American Society for Information Science* **32** (1981) 211-216.
8. C.S. Cater and D.H. Kraft, A generalization and clarification of the Waller-Kraft wish list, *Information Processing & Management* **25** (1989) 15-25.
9. F. Herrera and E. Herrera-Viedma, Aggregation operators for linguistic weighted information, *IEEE Transactions on Systems, Man, and Cybernetics, Part A,* **27** (1997) 646-656.
10. E. Herrera-Viedma, Modeling the retrieval process for an information retrieval system using an ordinal fuzzy linguistic approach, *Journal of the American Society for Information Science and Technology* **52:6** (2001) 460-475.
11. D.H. Kraft and D.A. Buell, Fuzzy sets and generalized Boolean retrieval systems, *International Journal of Man-Machine Studies* **19** (1983) 45-56.
12. D.H. Kraft, G. Bordogna and G. Pasi, An extended fuzzy linguistic approach to generalize Boolean information retrieval, *Information Sciences* **2** (1994) 119-134.
13. S. Miyamoto, *Fuzzy Sets in Information Retrieval and Cluster Analysis* (Kluwer Academic Publishers, 1990).
14. G. Salton and C. Buckley, Term weighting approaches in automatic text retrieval, *Information Processing & Management* **24** (1988) 513-523.
15. W.G. Waller and D.H. Kraft, A mathematical model of a weighted Boolean retrieval system, *Information Processing & Management* **15** (1979) 235-245.
16. L.A. Zadeh, The concept of a linguistic variable and its applications to approximate reasoning. Part I, *Information Sciences* **8** (1975) 199-249, Part II, *Information Sciences* **8** (1975) 301-357, Part III, *Information Sciences* **9** (1975) 43-80.