International Journal of Uncertainty,
Fuzziness and Knowledge-Based Systems
Vol. 15, No. 2 (2007) 225–250

A FUZZY LINGUISTIC IRS MODEL BASED ON A 2-TUPLE FUZZY LINGUISTIC APPROACH*

E. HERRERA-VIEDMA
Dept. of Computer Science and A.I., University of Granada, Granada, Spain
viedma@decsai.ugr.es

A. G. LÓPEZ-HERRERA
Dept. of Computer Science, University of Jaén, Jaén, Spain
aglopez@ujaen.es

M. LUQUE† and C. PORCEL‡
Dept. of Computer Science and N.A., University of Córdoba, Córdoba, Spain
†*mluque@uco.es*
‡*carlos.porcel@uco.es*

Received 2 July 2005
Revised 16 January 2007

Information Retrieval Systems (IRSs) based on an ordinal fuzzy linguistic approach present some problems of loss of information and lack of precision when working with discrete linguistic expression domains or when applying approximation operations in the symbolic aggregation methods. In this paper, we present a new IRS model based on the 2-tuple fuzzy linguistic approach, which allows us to overcome the problems of ordinal fuzzy linguistic IRSs and improve their performance.

Keywords: Information retrieval; fuzzy linguistic modelling; weighted queries.

1. Introduction

The main activity of an Information Retrieval System (IRS) is the gathering of pertinent archived documents that better satisfy the user queries. IRSs present three components to carry out this activity[1,2]:

(1) A documentary archive which stores the documents and the representation of their information contents (index terms).
(2) A query component which allows users to formulate their queries by means of a query language.

*This research has been supported by projects TIC2003-07977 and TIC-00602.

(3) A query evaluation component which evaluates the documents for a user query obtaining a Retrieval Status Value (RSV) for each document.

The query component 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.[3–12] Usually, they assume numeric weights associated with the queries (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[1,2,13–16] have been proposed using a fuzzy linguistic approach[17–19] 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 linguistic IRSs[1,2,14,15] is called the ordinal fuzzy linguistic approach.[20–24] In this approach, the query weights and document scores are ordered linguistic terms. These models of IRSs are affected by the two characteristic problems of ordinal fuzzy linguistic modelling[25,26]:

- The loss of precision: The ordinal fuzzy linguistic approach works with discrete linguistic domains and this implies some limitations in the representation of the linguistic information, e.g. to represent the relevance degrees.
- The loss of information: Aggregation operators of ordinal linguistic information use approximation operations in their definitions (e.g. rounding operation), and thus this produces the consequent loss of information.

In[1] we presented an ordinal fuzzy linguistic IRS that accepts weighted queries based only on one weighting levels (query terms) and allows to associate different semantic interpretations to the weights. Its query language is based on a Boolean query language and it uses the t-conorm Max and t-norm Min as operators to evaluate the Boolean logical connectives OR and AND in the retrieval process. In[2] we extended that ordinal fuzzy linguistic IRS model[1] and we presented a new IRS model that allows users to associate linguistic weights on two weighting levels, query terms and query subexpressions. This new model uses the same operators to model the Boolean logical connectives OR and AND in the retrieval process. In[15] we presented an ordinal fuzzy linguistic IRS model which allows to represent the different information concepts (importance, relevance) that appear in a retrieval process with different linguistic term sets, that is, using multi-granular linguistic contexts. All these models[1,2,15] present the aforementioned limitations associated to the use of ordinal fuzzy linguistic information, loss of information and lack of precision, as

well as the loss of flexibility in the computation of the RVSs of documents due to the use of the operators Max and Min.

The main aim of this paper is to present a new model of a fuzzy linguistic IRS based on the 2-tuple fuzzy linguistic approach,[25] whose application on the representation of linguistic information allows us to overcome the main limitations of the ordinal fuzzy linguistic IRS models.[1,2,15] The 2-tuple fuzzy linguistic modelling solves the problems of ordinal one (loss of information and lack of precision). Furthermore, we introduce a new soft computing operator to model the Boolean connectives in a more flexible way, the 2-tuple linguistic LOWA (Linguistic Ordered Weighted Averaging) operator. In such a way, we improve the performance of previous ordinal fuzzy linguistic IRS models[1,2,15] with a limited cost, and it could contribute to increase the users' degree of satisfaction.

The paper is set out as follows. In Section 2 the preliminaries on the ordinal fuzzy linguistic approach, on an ordinal fuzzy linguistic IRS model and on the 2-tuple fuzzy linguistic approach are presented. The new 2-tuple fuzzy linguistic IRS model is defined in Section 3. Finally, Section 4 draws our conclusions.

2. Preliminaries

In this section we present the basic elements needed to understand our new proposal: the ordinal fuzzy linguistic approach,[21] the ordinal fuzzy linguistic IRS model defined in,[1] and the 2-tuple fuzzy linguistic approach.[25]

2.1. *The ordinal fuzzy linguistic approach*

The fuzzy linguistic approach is an approximate tool used to model qualitative information in a problem. It is based on the concept of linguistic variable and has been satisfactorily used in many problems.[13,21,27–33]

The ordinal fuzzy linguistic approach[21] is a type of fuzzy linguistic approach. An ordinal fuzzy linguistic approach is defined by considering a finite and totally ordered label set $\mathcal{S} = \{s_0, \ldots, s_T\}$, $T+1$ being the cardinality of $\mathcal{S}$ in the usual sense, and with odd cardinality (usually 7 or 9 labels). It is also assumed that each linguistic label s_i has assigned a triangular membership function μ_{s_i} represented by three parameters, (a_i, b_i, c_i), being b_i the central point of the function, and a_i and c_i the left and right points, respectively.

Example 1. A set with 7 linguistic labels could be that drawn in Figure 1 with $\mathcal{S} = \{s_0 = Null(N),\ s_1 = Very\_Low(VL),\ s_2 = Low(L),\ s_3 = Medium(M),\ s_4 = High(H),\ s_5 = Very\_High(VH),\ s_6 = Total(TO)\}$, and the following triangular membership functions

$$\begin{array}{ll} N = (0, 0, .17) & VL = (0, .17, .33) \\ L = (.17, .33, .5) & M = (.33, .5, .67) \\ H = (.5, .67, .83) & VH = (.67, .83, 1) \\ TO = (.83, 1, 1). & \end{array}$$

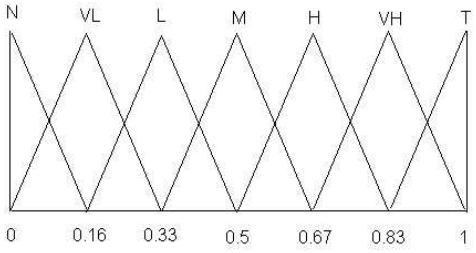

Fig. 1. A set with 7 labels.

Assuming the unit interval $[0, 1]$ as reference domain, the mid term (M) represents an assessment of "approximately 0.5" and the rest of the terms are placed symmetrically around it.[34]

The semantics of the linguistic terms set is established from the ordered structure of the terms set by considering that each linguistic term for the pair (s_i, s_{T-i}) is equally informative.

The computational model to combine ordinal linguistic information consists of three types of operators:

(1) Negation operator: $Neg(s_i) = s_j, j = T - i$.
(2) Comparison operators:
   - Maximization operator: $MAX(s_i, s_j) = s_i$ if $s_i \geq s_j$.
   - Minimization operator: $MIN(s_i, s_j) = s_i$ if $s_i \leq s_j$.
(3) Aggregation operators: Usually to combine ordinal linguistic information we use aggregation operators based on symbolic computation, e.g. the LOWA operator,[21] which is a linguistic OWA operator[35] defined using the convex combination of linguistic labels.[36]

Definition 1. .[21] Let $\{a_1, \ldots, a_m\}$ be a set of labels to aggregate, then the **LOWA** operator ϕ is defined as:

$$\phi(a_1, \ldots, a_m) = W \cdot B^T = C^m\{w_k, b_k, k = 1, \ldots, m\} =$$

$$= w_1 \otimes b_1 \oplus (1 - w_1) \otimes C^{m-1}\{\beta_h, b_h, h = 2, \ldots, m\},$$

where $W = [w_1, \ldots, w_n]$, is a weighting vector, such that, $w_1 \in [0, 1]$ and $\sum_i w_i = 1$, $\beta_h = \frac{w_h}{\sum_2^m w_k}, h = \{2, \ldots, m\}$, and B is the associated ordered label vector. Each element $b_i \in B$ is the i-th largest label in the collection $\{a_1, \ldots, a_m\}$, and C^m is the convex combination operator of m labels. If $w_j = 1$ and $w_i = 0$ with $i \neq j \forall i$, the combination is defined as: $C^m\{w_i, b_i, i = 1, \ldots, m\} = b_j$.
And if $m = 2$ then it is defined as:

$$C^2\{w_i, b_i, i = 1, 2\} = w_1 \otimes s_j \oplus (1 - w_1) \otimes s_i = s_k, s_j, s_i \in \mathcal{S}, (j \geq i),$$

such that $k = min\{T, i + round(w_1 \cdot (j - i))\}$, where $round(\cdot)$ is the usual round operation, and $b_1 = s_j$, $b_2 = s_i$.

We should point out that the LOWA operator presents two important advantages with respect to other linguistic aggregation operators. Firstly, it allows to aggregate linguistic information in an automatic way and no linguistic approximation process is necessary, and secondly, it allows to soften the hard behaviour of the usual fuzzy connectives, t-norms and t-conorms, which is very useful in particular applications, as in IR to model the evaluation of the Boolean connectives. This last one is illustrated in the following example.

Example 2. Suppose $m = 3$, $W = [.7, .2, .1]$ and the label set with 9 labels $\{s_0 = N, s_1 = EL, s_2 = VL, s_3 = L, s_4 = M, s_5 = H, s_6 = VH, s_7 = EH, s_8 = TO\}$. So, if we aggregate the linguistic values $\{TO, VL, EL\}$ with the linguistic t-conorm MAX, the result clearly is the label TO, whereas if we use the LOWA operator the result is VH, which is computed as follows:

$$\phi(TO, VL, EL) = C^3\{(.7, TO), (.2, VL), (0.1, EL)\} =$$

$$.7 \otimes TO \oplus .3 \otimes C^2\{(.66, VL), (.34, EL)\}$$

As $C^2\{(.66, VL = s_2), (.34, EL = s_1)\} = s_2 = VL$ because $Min\ \{8,\ 1 + round((2 - 1) \cdot .66)\} = Min\{8,\ 2\} = 2$, then $\phi(TO, VL, EL) = VH$ given that $Min\ \{8,\ 2 + round((8 - 2) \cdot .7)\} = Min\{8,\ 6\} = 6$, and $s_6 = VH$.

2.2. *An ordinal fuzzy linguistic IRS model*

In,[1] we proposed an ordinal linguistic weighted IRS that presents the following elements to carry out its activity:

Documentary archive. This archive stores the finite set of documents $\mathcal{D} = \{d_1, \ldots, d_m\}$ represented by a finite set of index terms $\mathcal{T} = \{t_1, \ldots, t_l\}$, 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 $\mathcal{F} : \mathcal{D} \times \mathcal{T} \to [0, 1]$, which is called index term weighting function:[10]

$$d_j = \mathcal{F}(d_j, t_1)/t_1 + \mathcal{F}(d_j, t_2)/t_2 + \ldots + \mathcal{F}(d_j, t_l)/t_l.$$

$\mathcal{F}$ weighs index terms according to their significance in describing the content of a document. Thus $\mathcal{F}(d_j, t_i)$ is a numerical weight that represents the degree of significance of t_i in d_j.

Query component. Query component is based on a weighted Boolean query language to express user information needs. As it is known the Boolean query language is used in both Boolean and extended Boolean IR models. With this language each query is expressed as a combination of the weighted index terms that are connected by logical operators AND ($\wedge$), OR ($\vee$), and NOT ($\neg$). This query component allows users to weigh each term in a query according to three

different semantics possibilities, which could be used simultaneously by them with enough knowledge. As in,[13] we used the linguistic variable *Importance* to express the linguistic weights associated to the query terms. Thus, we considered a set of ordinal linguistic values $\mathcal{S}$ to express the linguistic weights. Then, we defined a linguistic weighted Boolean query as any legitimate Boolean expression whose atomic components (atoms) are quadruples $< t_i, c_i^1, c_i^2, c_i^3 >$ belonging to the set, $\mathcal{T} \times \mathcal{S}^3, t_i \in T$, and c_i^1, c_i^2, c_i^3 are ordinal values of the linguistic variable *Importance*, modelling a symmetrical threshold semantics, a quantitative semantics, and a relative importance semantics, respectively. Accordingly, the set $\mathcal{Q}$ of the legitimate queries is defined by the following syntactic rules:

(1) $\forall q =< t_i, c_i^1, c_i^2, c_i^3 > \in \mathcal{T} \times \mathcal{S}^3 \rightarrow q \in \mathcal{Q}$.
(2) $\forall q, p \in \mathcal{Q} \rightarrow q \wedge p \in \mathcal{Q}$.
(3) $\forall q, p \in \mathcal{Q} \rightarrow q \vee p \in \mathcal{Q}$.
(4) $\forall q \in \mathcal{Q} \rightarrow \neg q \in \mathcal{Q}$.
(5) All legitimate queries $q \in \mathcal{Q}$ are only those obtained by applying rules 1-4, inclusive.

We should point out that although the three semantics could be used simultaneously, this is very difficult, even for expert users. We assume that when a user wants to provide his information needs with our language he previously has to decide how many semantics to use. The important aspect of this language is that it generalizes those languages based on only one weighting semantics, allowing us to express our information needs by choosing among three possibilities to weigh query terms.

Query evaluation component. The goal of the evaluation component is to evaluate documents in terms of their relevance to a linguistic weighted Boolean query according to the above three possible semantics. A Boolean query with more than one weighted term is evaluated by means of a constructive bottom-up process based on the criterion of separability.[8,10] This process includes the five subsequent steps:

(1) *Preprocessing of the query:* In this step, the user query is preprocessed to put it into either conjunctive normal form (CNF) or disjunctive normal form (DNF), with the result that all its Boolean subexpressions must have more than two atoms.
(2) *Evaluation of atoms with respect to the symmetrical threshold semantics:* In this step, the documents are evaluated with regard to their relevance to individual atoms in the query, considering only the restrictions imposed by the symmetrical threshold semantics. With a usual threshold semantics[16] a weighted term expresses the minimally acceptable documents for a user, that is, a query $< t_i, w_i >$ is synonymous with the query $< t_i,$ "*at least* $w_i'' >$, and such an interpretation is modelled by a non-decreasing matching function. However, in[1] we assumed that a user can search for documents with a minimally acceptable presence of one term or

documents with a maximally acceptable absence of one term, and then we defined the symmetrical threshold semantics. 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 $\mathcal{F}$ exceed the established thresholds with a high RSV, but allowing some small partial credit for a document whose $\mathcal{F}$ values are lower than the thresholds. Then, the query weights indicate presence requirements, i.e., they are presence weights. *Symmetrical threshold semantics*[1,2] is a special threshold semantics 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).

Assuming this threshold semantics 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 an $\mathcal{F}$ 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 an $\mathcal{F}$ value less than *Low*. Given a request $< t_i, w_i^1, -, - >$, this means that the linguistic query weights that imply the presence of a term in a document $w_i^1 \geq s_{\mathcal{T}/2}$ (e.g. *High, Very High,*) must be treated differently to the linguistic query weights that imply the absence of one term in a document $w_i^1 < s_{\mathcal{T}/2}$ (e.g. *Low, Very Low*). Then, if $w_i^1 \geq s_{\mathcal{T}/2}$ the request $< t_i, w_i^1, -, - >$, is synonymous with the request $< t_i, at\ least\ w_i^1, -, - >$, which expresses the fact that the desired documents are those having $\mathcal{F}$ values as high as possible; and if $w_i^1 < s_{\mathcal{T}/2}$ then it is synonymous with the request $< t_i, at\ most\ w_i^1, -, - >$, which expresses the fact that the desired documents are those having $\mathcal{F}$ values as low as possible. This interpretation is modelled by the following linguistic matching function g^1 :

$$RSV_j^{i,1} = g^1(d_j, t_i, c_i^1) = \begin{cases} s_0 & s_b \geq s_{\frac{T}{2}} \wedge s_a = s_0 \\ s_{i_1} & s_b \geq s_{\frac{T}{2}} \wedge s_0 < s_a < s_b \\ s_{i_2} & s_b \geq s_{\frac{T}{2}} \wedge s_b \leq s_a < s_T \\ s_T & s_b \geq s_{\frac{T}{2}} \wedge s_a = s_T \\ s_T & s_b < s_{\frac{T}{2}} \wedge s_a = s_0 \\ Neg(s_{i_1}) & s_b < s_{\frac{T}{2}} \wedge s_0 < s_a \leq s_b \\ Neg(s_{i_2}) & s_b < s_{\frac{T}{2}} \wedge s_b < s_a < s_T \\ s_0 & s_b < s_{\frac{T}{2}} \wedge s_a = s_T \end{cases} \quad (1)$$

such that: $i_1 = Max\{0, round(b - \frac{b-a}{k})\}$, $i_2 = Min\{T, round(b + \frac{a-b}{k})\}$, $k \in \{1, 2, \ldots, b\}$ being a sensitivity parameter defined to control the importance of the closeness between $Label(\mathcal{F}(d_j, t_i))$ and c_i^1 in the final result. The greater the value of k, the smaller the importance of the value of distance. k affects the threshold fuzziness, and therefore, different k values can allow us to model different interpretations of the threshold semantics. g^1 was based on the distance or closeness between the linguistic index weight $Label(\mathcal{F}(d_j, t_i)) = s_a$ and the linguistic query term weight $c_i^1 = s_b$, being $Label : [0, 1] \rightarrow \mathcal{S}$ a function that assigns a label in $\mathcal{S}$ to a numeric value $r \in [0, 1]$ according to the following expression:

$$Label(r) = \mathrm{Sup}_q\{s_q \in S : \mu_{s_q}(r) = \mathrm{Sup}_v \{\mu_{s_v}(r)\}\}.$$

(3) *Evaluation of atoms with respect to the quantitative semantics:* In this step, the documents are evaluated with regard to their relevance to individual atoms of query, but this time, considering the restrictions imposed by the quantitative semantics. In,[1] the evaluation of the atom $< t_i, c_i^1, c_i^2, c_i^3 >$ with respect to the quantitative semantics associated with c_i^2 for a document d_j, called $RSV_j^{i,1,2} \in \mathcal{S}$, was obtained by means of the linguistic matching function $g^2 : \mathcal{D} \times \mathcal{S}^2 \rightarrow \mathcal{S}$ as follows:

$$RSV_j^{i,1,2} = g^2(d_j, RSV_j^{i,1}, c_i^2) = \begin{cases} s_0 & d_j \notin \beta^S \\ RSV_j^{i,1} & d_j \in \beta^S \end{cases} \tag{2}$$

where β^S is the set of documents such that $\beta^S \subseteq \mathrm{Supp}(M_i)$ where $M_i = \{(d_1, RSV_1^{i,1}), \ldots, (d_m, RSV_m^{i,1})\}$, is a fuzzy subset of documents obtained according to the followings steps:

(a) $K = \sharp\mathrm{Supp}(M_i)$.
(b) REPEAT
$M_i^K = \{s_q \in \mathcal{S} : \mu_{s_q}(\frac{K}{m}) = \mathrm{Sup}_v\{\mu_{s_v(\frac{K}{m})}\}\}$.
$S^K = \mathrm{Sup}_q\{s_q \in M_i^K\}$.
$K = K - 1$.
(c) UNTIL$((c_i^2 \in M_i^{K+1}) \vee (c_i^2 \geq S^{K+1}))$.
(d) $\beta^S = \{d_{\sigma(1)}, \ldots, d_{\sigma(K+1)}\}$, such that $RSV_{\sigma(h)}^{i,1} \leq RSV_{\sigma(l)}^{i,1}, \forall l \leq h$.

According to g^2, the application of the quantitative semantics reduces the number of documents to be considered in the evaluation of t_i in the later steps.

(4) *Evaluation of subexpressions and modelling the importance semantics:* In this step, the documents are evaluated with regards to their relevance to Boolean subexpressions of the queries (Boolean combinations of atoms established by means of the logical connectives), considering the restrictions imposed on the connected atoms by the importance semantics. We may have two kinds of subexpressions:conjunctive, or disjunctive ones. To model the connective AND we use the linguistic MIN operator and to

model the connective OR we use the linguistic MAX operator. In case of connective AND, the evaluation of importance weights is introduced by using the linguistic transformation function $MAX(Neg(weight), value)$, and in the OR connective case by using the linguistic transformation function $MIN(weight, value)$.

(5) *Evaluation of the whole query:* In this final step of evaluation, the documents are evaluated with regards to their relevance to Boolean combinations in all the Boolean subexpressions existing in a query. To evaluate the connectives AND and OR we use the operators linguistic MIN and MAX, respectively.

2.3. *The 2-tuple fuzzy linguistic model*

Let $\mathcal{S} = \{s_0, \ldots, s_T\}$ be a linguistic term set, if a symbolic method aggregating linguistic information obtains a value $\beta \in [0, T]$, and $\beta \notin \{0, \ldots, T\}$ then an approximation function ($app(\cdot)$) is used to express the index of the result in $\mathcal{S}$.[25] For example, in the LOWA defined in Section 2.1, $app(\cdot)$ is the simple function *round*.

Definition 2. Let $\beta \in [0, T]$ be the result of an aggregation of the indexes of a set of labels assessed in a linguistic term set $\mathcal{S}$, i.e., the result of a symbolic aggregation operation. Let $i = round(\beta)$ and $\alpha_i = \beta - i$ be two values, such that, $i \in \{0, 1, \ldots, T\}$ and $\alpha_i \in [-.5, .5)$ then α_i is called a Symbolic Translation.

Roughly speaking, the symbolic translation of a linguistic term, s_i, is a numerical value assessed in $[-.5, .5)$ that supports the "difference of information" between a information value $\beta \in [0, T]$ obtained after a symbolic aggregation operation and the closest value in $\{0, \ldots, T\}$ that indicates the index of the closest linguistic term in $\mathcal{S}$ ($i = round(\beta)$).

From the concept of symbolic translation, Herrera and Martínez developed a linguistic representation model which represents the linguistic information by means of 2-tuples (s_i, α_i), $s_i \in \mathcal{S}$ and $\alpha_i \in [-.5, .5)$: i) s_i represents the linguistic label of the information, and ii) α_i is a numerical value expressing the value of the translation from the original result β to the closest index label i in $\mathcal{S}$.

This model presents a set of transformation functions between numeric values and linguistic 2-tuples.

Definition 3. [25] Let $\mathcal{S}$ be a linguistic term set and $\beta \in [0, T]$, then the 2-tuple that expresses the equivalent information to β is obtained with the following function:

$$\Delta : [0, T] \to \mathcal{S} \times [-.5, .5),$$

$$\Delta(\beta) = (s_i, \alpha_i), with \begin{cases} s_i & i = round(\beta) \\ \alpha = \beta - i & \alpha \in [-.5, .5) \end{cases} \tag{3}$$

where s_i has the closest index label to β and α_i is the value of the symbolic translation (Figure 2).

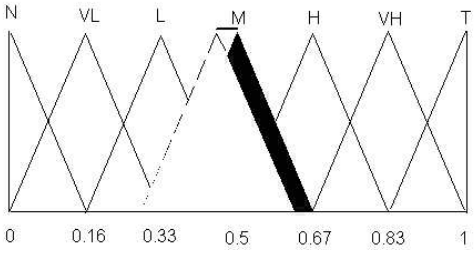

Fig. 2. Symbolic translation.

Example 3. Let us suppose a symbolic aggregation operation over labels assessed in $\mathcal{S} = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6\}$ that obtains as its result $\beta = 2.8$, then the representation of this information by means of a linguistic 2-tuple will be:

$$\Delta(2.8) = (s_3, -.2)$$

Proposition 1. [25] *Let (s_i, α_i), $s_i \in \mathcal{S}$ be a linguistic 2-tuple. There exists always a Δ^{-1} function, such that, from a 2-tuple it returns its equivalent numerical value $\beta \in [0, T] \subset \mathbb{R}$.*

Remark 1. [25] From Definition 3 and Proposition 1, it is obvious that the conversion of a linguistic term into a linguistic 2-tuple consists of adding a value 0 as symbolic translation: $s_i \in \mathcal{S} \rightarrow (s_i, 0)$.

The 2-tuple linguistic computational model operates with the 2-tuples without loss of information and is based on the following operations:[25]

(1) Negation operator of a 2-tuple: $NEG(s_i, \alpha_i) = \Delta(T - \Delta^{-1}(s_i, \alpha_i))$.
(2) Comparison of 2-tuples: The comparison of linguistic information represented by 2-tuples is carried out according to an ordinary lexicographic order. Let (s_k, α_1) and (s_l, α_2) be two linguistic 2-tuples:
   - if $k < l$ then (s_k, α_1) is smaller than (s_l, α_2).
   - if $k = 1$ then:
     (a) if $\alpha_1 = \alpha_2$ then (s_k, α_1), (s_l, α_2) represents the same information.
     (b) if $\alpha_1 < \alpha_2$ then (s_k, α_1) is smaller then (s_l, α_2).
     (c) if $\alpha_1 > \alpha_2$ than (s_k, α_1) is bigger than (s_l, α_2).
(3) Aggregation of 2-tuples: Using the functions Δ and Δ^{-1} any numerical aggregation operator can be easily extended for dealing with linguistic 2-tuples. Some examples are presented in.[25]

3. A 2-Tuple Fuzzy Linguistic IRS Model

In this section, we present a new fuzzy linguistic IRS model based on the 2-tuple fuzzy linguistic approach whose application to the representation of linguistic

information allows us to overcome the problems detected in.[1] The main novelty of this new linguistic IRS model is in the design of its query evaluation component that uses the advantages of the 2-tuple fuzzy linguistic model to avoid the loss of information and lack of precision. Furthermore, it includes a new soft computation operator, the 2-tuple LOWA operator, which is used to model the logical connectives AND and OR in a more flexible way.

In the following subsections, we introduce the query evaluation component of this new IRS model and analyze its performance.

3.1. *The query evaluation component of the 2-tuple fuzzy linguistic IRS model*

To define the query evaluation component we assume that users use the same query language presented in Subsection 2.2. Therefore, users use multi-weighted linguistic Boolean queries to express their information needs with weights which are assessed using usual ordinal linguistic terms. Furthermore, the underlying procedure of this new query evaluation component is similar to that presented in Subsection 2.3, that is, the evaluation of user queries is also carried out by means of a constructive bottom-up process based on the criterion of separability[10] and at the same time as supporting all the possible semantics of query weights considered. We should point out that the system allows the simultaneous use of all semantics, but really this is very difficult for a usual user. Really, the most important quality is that this type of multi-weighted query language increases user-system interaction, because for example, it allows a user to carry out different query sessions with different semantics depending on his needs.

In what follows we show the evaluation steps of this new query evaluation component, which are defined using the 2-tuple fuzzy linguistic approach.

(1) *Preprocessing of the query*
As in Subsection 2.3, the user query is preprocessed and put into either CNF or DNF (see Figure 3). We should point out that a user does not use the fuzzy linguistic 2-tuple representation to provide his information needs by means of linguistic weighted queries, he uses the ordinal fuzzy linguistic approach which is easier. The fuzzy linguistic 2-tuple representation is used in the evaluation of the queries to improve the results.

(2) *Evaluation of atoms with respect to the symmetrical threshold semantics*
In this step the documents are evaluated according to their relevance only to atoms of the query, by applying the symmetrical threshold semantics presented in Subsection 2.3 but defined in a 2-tuple fuzzy linguistic context. The matching function g^1 using now the 2-tuple linguistic representation model is called g^1_{2t} : $\mathcal{D} \times \mathcal{T} \times \mathcal{S} \to \mathcal{S} \times [-.5, .5))$.
Then, given an atom $< t_i, c_i^1, c_i^2, c_i^3 >$ and a document $d_j \in \mathcal{D}$, g^1_{2t} the linguistic RSV of d_j, called RSV_j, is obtained by measuring how well the index term weight $\mathcal{F}(d_j, t_i)$ satisfies the request expressed by the linguistic weight c_i^1

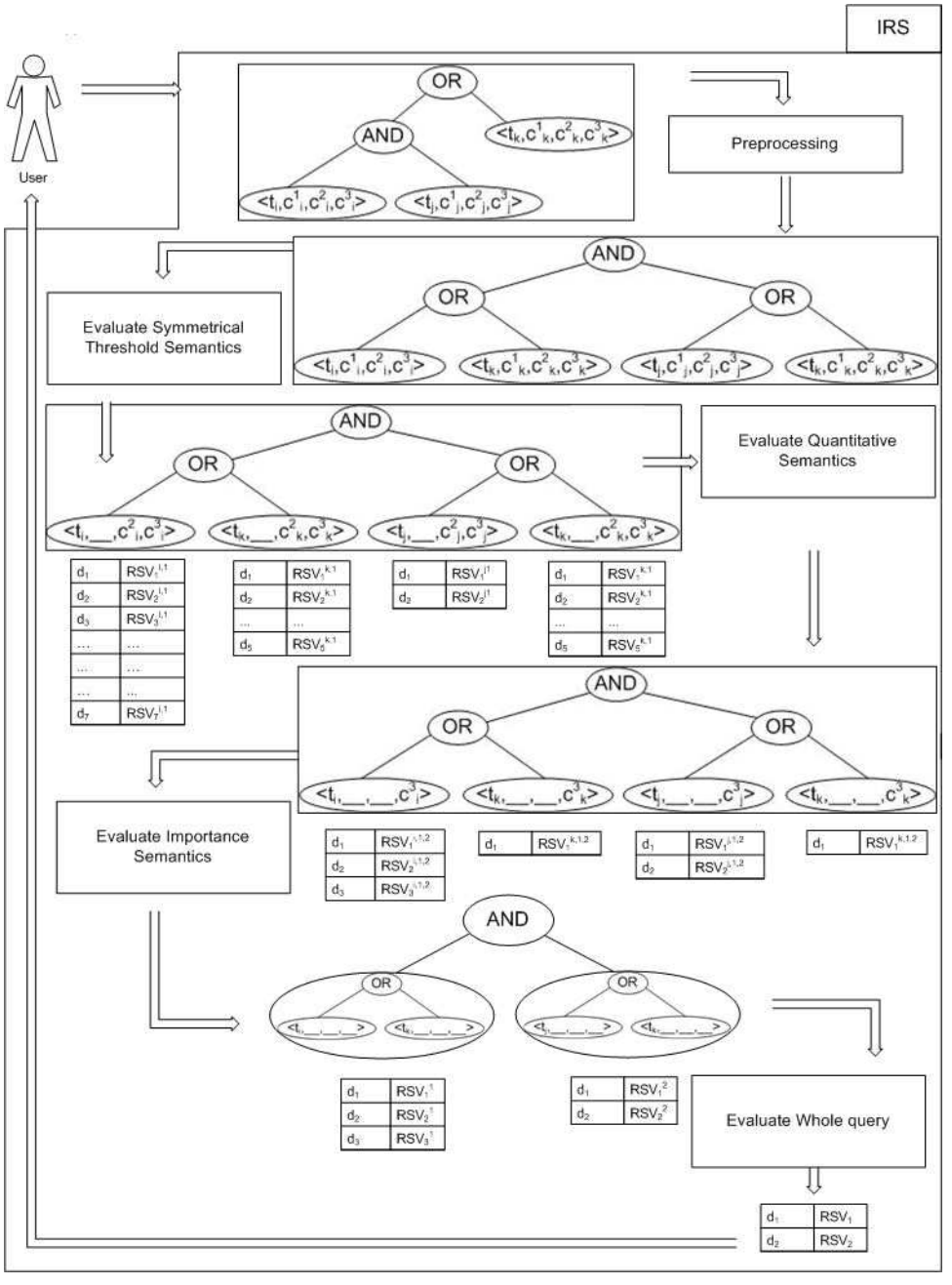

Fig. 3. Information retrieval process.

according to the following expression:

$$RSV_j^{i,1} = g_{2t}^1(d_j, t_i, c_i^1) = \begin{cases} (s_0,0) & (s_b,0) \geq (s_{\frac{T}{2}},0) \wedge (s_a,\alpha_a) = (s_0,0) \\ i_1 & (s_b,0) \geq (s_{\frac{T}{2}},0) \wedge (s_0,0) < (s_a,\alpha_a) < (s_b,0) \\ i_2 & (s_b,0) \geq (s_{\frac{T}{2}},0) \wedge (s_b,0) \leq (s_a,\alpha_a) < (s_T,0) \\ (s_T,0) & (s_b,0) \geq (s_{\frac{T}{2}},0) \wedge (s_a,\alpha_a) = (s_T,0) \\ (s_T,0) & (s_b,0) < (s_{\frac{T}{2}},0) \wedge (s_a,\alpha_a) = (s_0,0) \\ Neg(i_1) & (s_b,0) < (s_{\frac{T}{2}},0) \wedge (s_0,0) < (s_a,\alpha_a) \leq (s_b,0) \\ Neg(i_2) & (s_b,0) < (s_{\frac{T}{2}},0) \wedge (s_b,0) < (s_a,\alpha_a) < (s_T,0) \\ (s_0,0) & (s_b,0) < (s_{\frac{T}{2}},0) \wedge (s_a,\alpha_a) = (s_T,0) \end{cases} \quad (4)$$

such that: $i_1 = \Delta(\Delta^{-1}(s_b,0)) - \frac{\Delta^{-1}(s_b,0)-\Delta^{-1}(s_a,\alpha_a)}{k}$, $i_2 = \Delta(\Delta^{-1}(s_b,0) + \frac{\Delta^{-1}(s_a,\alpha_a)-\Delta^{-1}(s_b,0)}{k})$, $k \in \{1,2,\ldots,b\}$, $(s_a,\alpha_a) = \Delta(T \cdot \mathcal{F}(d_j,t_i))$ and $(s_b,0)$ is the threshold value c_i^1 in the 2-tuple linguistic representation approach.

(3) *Evaluation of atoms with respect to the quantitative semantics*

In this step, documents are evaluated with regard to their relevance to individual atoms of the query, but considering the restrictions imposed by the quantitative semantics. The linguistic quantitative weights are interpreted as follows:[1] when a user establishes a certain quantity of documents for a term in the query, expressed by a linguistic quantitative weight, then the set of documents to be retrieved must have the minimum number of documents that satisfies the compatibility or membership function associated with the meaning of the label used as linguistic quantitative weight. Furthermore, these documents must be those that better satisfy the threshold restrictions imposed on the term.

Therefore, given an atom $< t_i, c_i^1, c_i^2, c_i^3 >$ and assuming that $RSV_j^{i,1} \in (\mathcal{S} \times [-.5,.5))$ represents the evaluation according to the symmetrical threshold semantics for d_j, we model the interpretation of a quantitative semantics by means of a 2-tuple linguistic matching function, called g_{2t}^2. This function is defined between the $RSV_j^{i,1}$ and the linguistic quantitative weight $c_i^2 \in \mathcal{S}^2$. The evaluation value of the atom $< t_i, c_i^1, c_i^2, c_i^3 >$ with respect to c_i^2 for a document d_j, called $RSV_j^{i,1,2} \in (\mathcal{S} \times [-.5,.5))$, is obtained by means of the linguistic matching function $g_{2t}^2 : \mathcal{D} \times (\mathcal{S} \times [-.5,.5)) \times \mathcal{S} \rightarrow (\mathcal{S} \times [-.5,.5))$ defined according to the following expression:

$$RSV_j^{i,1,2} = g_{2t}^2(d_j, RSV_j^{i,1}, c_i^2) = \begin{cases} (s_0,0) & d_j \notin \beta^S \\ RSV_j^{i,1} & d_j \in \beta^S \end{cases} \quad (5)$$

β^S is a subset of documents obtained according to the following steps:

(a) $K = \sharp \text{Supp}(M_i)$.
(b) REPEAT
    $S^K = (s_e, \alpha_e) = \Delta(T \cdot \frac{K}{m})$.
    $K = K - 1$.
(c) UNTIL$((s_i^2, 0) \geq S^{K+1})$
(d) $\beta^S = \{d_{\sigma(1)},\ldots,d_{\sigma(K+1)}\}$, such that $RSV_{\sigma(h)}^{i,1} \leq RSV_{\sigma(l)}^{i,1}, \forall l \leq h$.

(4) *Evaluation of subexpressions and modelling of the relative importance semantics*
Then, in this step we have to evaluate the relevance of documents with respect to all subexpressions of preprocessed queries which are composed of a minimum number of two atomic components according to the application of preprocessing step.
Given a subexpression q_v with $\eta \geq 2$ atoms, we know that each document d_j presents a partial $RSV_j^{i,1,2} \in (\mathcal{S} \times [-.5, .5))$ with respect to each atom $< t_i, c_i^1, c_i^2, c_i^3 >$ of q_v. Then, the evaluation of the relevance of a document d_j with respect to the whole subexpression q_v implies the aggregation of the partial relevance degrees $\{RSV_j^{i,1,2}, i = 1, \ldots, \eta\}$ weighted by means of the respective relative importance degrees $\{c_i^3 \in \mathcal{S}, i = 1, \ldots, \eta\}$. To do that, we need a weighted aggregation operator of 2-tuple linguistic information which should guarantee that the more important the query terms, the more important they are in the determination of the RSVs.
Usually, a weighted aggregation operator to aggregate information carries out two activities:[20]

(a) The transformation of the weighted information under the importance degrees by means of a transformation function h; and
(b) The aggregation of the transformed weighted information by means of an aggregation operator of non-weighted information f. As it is known, the choice of h depends upon f.

In,[12] Yager discussed the effect of the importance degrees on the MAX and MIN types of aggregation and suggested a class of functions for importance transformation in both types of aggregation. For the MIN aggregation, he suggested a family of t-conorms acting on the weighted information and the negation of the importance degree, which presents the non-increasing monotonic property in these importance degrees. For the MAX aggregation, he suggested a family of t-norms acting on weighted information and the importance degree, which presents the non-decreasing monotonic property in these importance degrees.
Following Yager's recommendations, in[1] we proposed to model the conjunctive subexpressions by means of the linguistic t-norm MIN and transforming the weighted information under the importance degrees by means of the linguistic implication function MAX(NEG(weight),value), and the disjunctive subexpressions by means of the linguistic t-conorm MAX and transforming the weighted information under the importance degrees by means of the linguistic t-norm MIN. However, as it is known the evaluation of the logical connectives AND and OR by means of the MIN and MAX operators presents some limitations. That is, it may cause a very restrictive and inclusive behaviour, respectively. The problem is that the retrieval process may be deceptive because, on the one hand, the linguistic MIN t-norm may cause the rejection of useful documents by the dissatisfaction of any one single criterion of the conjunctive subexpression and, on the other hand, the linguistic MAX t-conorm may cause the acceptance

of a useless document by the satisfaction of any single criterion.
Consequently, to aggregate 2-tuple linguistic information we define the 2-tuple LOWA operator ϕ_{2t}, which is an extension of the LOWA operator ϕ presented in Subsection 2.1. Furthermore, this new operator allows to model both Boolean connectives AND and OR, and it overcomes the above limitations of the linguistic t-norm MIN and t-conorm MAX because its behaviour can be soften by means of the weighting vector.

Definition 4. Let $\{(a_1, \alpha_1), \ldots, (a_m, \alpha_m)\}$ be a set of 2-tuple assessments to aggregate, then the LOWA_{2t} operator ϕ_{2t} is definition as:

$$\phi_{2t}((a_1, \alpha_1), \ldots, (a_m, \alpha_m)) = W \cdot B^T = C_{2t}^m\{w_k, b_k, k = 1, \ldots, m\} =$$

$$w_1 \otimes b_1 \oplus (1 - w_1) \otimes C_{2t}^{m-1}\{\beta_h, b_h, h = 2, \ldots, m\}$$

where $b_i = (a_i, \alpha_i) \in (\mathcal{S} \times [-.5, .5))$, $W = [w_1, \ldots, w_m]$ is a weighting vector, such that $t_i \in [0, 1]$ and $\sum_i w_i = 1$, $\beta_h = \frac{w_h}{\sum_2^m w_k}$, $h = 2, \ldots, m$, and B is the associated ordered 2-tuple vector. Each element $b_i \in B$ is the i-th largest 2-tuple in the collection $\{(a_1, \alpha_1), \ldots, (a_m, \alpha_m))$, and C_{2t}^m is the convex combination operator of m 2-tuples. If $w_j = 1$ and $w_i = 0$ with $i \neq j \forall i, j$, the convex combination is defined as: $C_{2t}^m\{w_i, b_i, i = 1, \ldots, m\} = b_j$. And if $m = 2$ then it is defined as:

$$C_{2t}^2\{w_l, b_l, l = 1, 2\} = w_1 \otimes b_j \oplus (1 - w_1) \otimes b_i = \Delta(\lambda)$$

where $\lambda = \Delta^{-1}(b_i) + w_1 \cdot (\Delta^{-1}(b_j) - \Delta^{-1}(b_i))$, $b_j, b_i \in \mathcal{S} \times [-.5, .5)$, $(b_j \geq b_i)$, $\lambda \in [0, T]$.

In order to classify OWA operators in regards to their location between "and" and "or" Yager[35] introduced an orness measure associated with any vector W, which allows to characterize its aggregation behaviour:

$$orness(W) = \frac{1}{m-1} \sum_{i=1}^{m} (m - i) \cdot w_i.$$

Given a weighting vector W, then the closer an OWA operator is to an "or", the closer its orness measure is to one; while the nearer it is to an "and", the closer is the latter measure to zero. Generally, an OWA operator with much of the nonzero weights near the top will be an orlike operator ($orness(W) > 0.5$), and when the most of the nonzero weights are near the bottom, the OWA operator will be an andlike operator ($orness(W) \leq 0.5$). We use this good property in our linguistic IRS to evaluate the logical connectives of Boolean queries OR and AND.
Then, we use this orness measure to characterize the behaviour of the 2-tuple LOWA operators ϕ_{2t}. In particular, we propose to use a 2-tuple LOWA operator ϕ_{2t}^1 with $orness(W) \leq 0.5$ to model the AND connective and a 2-tuple LOWA operator ϕ_{2t}^2 with $orness(W) > 0.5$ to model the OR connective.

Hence, to evaluate the subexpressions together with the relative importance semantics and according to activities necessary to aggregate weighted information, if the subexpression is conjunctive then we use $f = \phi_{2t}^1$ and $h = MAX_{2t}(NEG(weight, 0), 2\text{-tuple\_value})$, and if it is disjunctive then we use $f = \phi_{2t}^2$, then $h = MIN_{2t}((weight, 0), 2\text{-tuple\_value})$, being MAX_{2t} and MIN_{2t} obtained according the comparison operation of 2-tuples defined in Subsection 2.2.

Shortly, given a document d_j, we evaluate its relevance with respect to a subexpression q_v, called $RSV_j^v \in (\mathcal{S} \times [-.5, .5))$ as:

(a) If q_v is a conjunctive subexpression then

$$RSV_j^v = \phi_{2t}^1(MAX_{2t}(Neg(c_1^3, 0), RSV_j^{1,1,2}), \ldots, MAX_{2t}(Neg(c_\eta^3, 0), RSV_j^{\eta,1,2})).$$

(b) If q_v is a disjunctive subexpression then

$$RSV_j^v = \phi_{2t}^2(MIN_{2t}((c_1^3, 0), RSV_j^{1,1,2}), \ldots, MIN_{2t}((c_\eta^3, 0), RSV_j^{\eta,1,2})).$$

(5) *Evaluation of the whole query*

In this step, the final evaluation of each document is achieved by combining their evaluations with respect to all the subexpressions. To do that, we use again both 2-tuple LOWA operators ϕ_{2t}^1 and ϕ_{2t}^2 to model the AND and OR connectives, respectively.

Then, given a document d_j, its relevance with respect to a query, $RSV_j \in (\mathcal{S} \times [-.5, .5))$, is obtained as:

(a) If q is in CNF then $RSV_j = \phi_{2t}^1(RSV_j^1, \ldots, RSV_j^v)$, and
(b) If q is in DNF then $RSV_j = \phi_{2t}^2(RSV_j^1, \ldots, RSV_j^v)$,

with v standing for the number of subexpressions of q.

This evaluation process of a query is shown in Figure 3.

Remark 2. On the NOT Operator. We should note that, if a query is in CNF or DNF, we have to define the negation operator only at the level of single atoms. This simplifies the definition of the NOT operator. As was done in,[1] the evaluation of document d_j for a negated weighted atom $< \neg t_i, c_i^1, c_i^2, c_i^3 >$ is obtained from the negation of the index term weight $\mathcal{F}(t_i, d_j)$. This means to calculate the threshold matching function g_{2t}^1 from the linguistic 2-tuple value $(s_a, \alpha_a) = \Delta(T \cdot (1 - \mathcal{F}(d_j, t_i)))$.

Shortly, this query evaluation component can be synthesized by means of a general linguistic evaluation function $\mathcal{E}_{2t} : \mathcal{D} \times \mathcal{Q} \rightarrow (\mathcal{S} \times [-.5, .5))$, which evaluates the different kind of preprocessed queries,$\{q =< t_i, c_i^1, c_i^2, c_i^3 >, q \wedge p, q \vee p, \neg q\}$ according to the following five rules:

(a) Atoms:

$$\mathcal{E}_{2t}(d_j, q^1) = g_{2t}^2(d_j, g_{2t}^1(d_j, t_i, c_i^1), c_i^2),$$

such that $q^1 =< t_i, c_i^1, c_i^2, c_i^3 >$.

(b) Conjunctive subexpressions:

$$\mathcal{E}_{2t}(d_j, q^2) = \phi^1_{2t}(MAX_{2t}(Neg(c^3_1, 0), \mathcal{E}_{2t}(d_j, q^1_1)), \ldots, MAX_{2t}(Neg(c^3_\eta, 0), \mathcal{E}_{2t}(d_j, q^1_\eta))),$$

being η the number of atoms of q^2.

(c) Disjunctive subexpressions:

$$\mathcal{E}_{2t}(d_j, q^3) = \phi^2_{2t}(MIN_{2t}((c^3_1, 0), \mathcal{E}_{2t}(d_j, q^1_1)), \ldots, MIN_{2t}((c^3_\eta, 0), \mathcal{E}_{2t}(d_j, q^1_\eta))).$$

(d) Query in CNF:

$$\mathcal{E}_{2t}(d_j, q^4) = \phi^1_{2t}(\mathcal{E}_{2t}(d_j, q^3_1), \ldots, \mathcal{E}_{2t}(d_j, q^3_\omega))$$

being ω the number of conjunctive subexpressions.

(e) Query in DNF:

$$\mathcal{E}_{2t}(d_j, q^5) = \phi^2_{2t}(\mathcal{E}_{2t}(d_j, q^2_1), \ldots, \mathcal{E}_{2t}(d_j, q^2_\omega))$$

being ω the number of conjunctive subexpressions.

Then, the result of system for any user query q is a fuzzy subset of documents characterized by the linguistic membership function $\mathcal{E}_{2t}$:

$$\{(d_1, \mathcal{E}_{2t}(d_1, q^k)), \ldots, (d_m, \mathcal{E}_{2t}(d_m, q^k))\}, k \in 1, 2, 3, 4, 5.$$

The documents are shown in decreasing order of $\mathcal{E}_{2t}$ 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 chosen for representing the linguistic variable *Relevance*.

3.2. *Operation of 2-tuple fuzzy linguistic weighted IRS*

In this subsection, we present an example of performance of the proposed IRS model. We also compare its performance with respect to the performance of the ordinal linguistic IRS model defined in.[1]

Let us suppose a small documentary archive containing a set of seven documents $\mathcal{D} = \{d_1, \ldots, d_7\}$, represented by means of a set of ten index terms $\mathcal{T} = \{t_1, \ldots, t_{10}\}$. Documents are indexed by means of a numeric indexing function $\mathcal{F}$, which represents them as follows:

$d_1 = 0.7/t_5 + 0.4/t_6 + 1/t_7$
$d_2 = 1/t_4 + 0.6/t_5 + 0.8/t_6 + 0.9/t_7$
$d_3 = 0.5/t_2 + 1/t_3 + 0.8/t_4$
$d_4 = 0.9/t_4 + 0.5/t_6 + 1/t_7$
$d_5 = 0.7/t_3 + 1/t_4 + 0.4/t_5 + 0.8/t_9 + 0.6/t_10$
$d_6 = 0.8/t_5 + 0.99/t_6 + 0.8/t_7$
$d_7 = 0.8/t_5 + 0.02/t_6 + 0.8/t_7 + 0.9/t_8$

Then, using the set of nine labels given in Example 2 and the 2-tuple transformation function Δ we obtain these documents in a 2-tuple 2-tuple linguistic representation:

$d_1 = (VH, -.4)/t_5 + (L, .2)/t_6 + (TO, 0)/t_7$
$d_2 = (TO, 0)/t_4 + (H, -.2)/t_5 + (VH, .4)/t_6 + (EH, .2)/t_7$
$d_3 = (M, 0)/t_2 + (TO, 0)/t_3 + (VH, .4)/t_4$
$d_4 = (EH, .2)/t_4 + (M, 0)/t_6 + (TO, 0)/t_7$
$d_5 = (VH, -.4)/t_3 + (TO, 0)/t_4 + (L, .2)/t_5 + (VH, .4)/t_9 + (H, -.2)/t_{10}$
$d_6 = (VH, .4)/t_5 + (TO, -.08)/t_6 + (VH, .4)/t_7$
$d_7 = (VH, .4)/t_5 + (N, .16)/t_6 + (VH, .4)/t_7 + (EH, .2)/t_8$

Suppose that a user formulates the following linguistic weighted query:

$$q = ((t_5, VH, VL, VH) \wedge (t_6, L, L, VL)) \vee (t_7, H, L, H).$$

Then, the evaluation of q is carried out in the following steps:

(1) *Preprocessing of the query*
The query q is in DNF, but it presents one subexpression with only one atom. Therefore, q must be preprocessed and transformed into a normal form with everyone of its subexpressions with a minimum number of two atoms. Then, q is transformed into the following equivalent query:

$$q' = ((t_5, VH, VL, VH) \vee (t_7, H, L, H)) \wedge ((t_6, L, L, VL) \vee (t_7, H, L, H)),$$

which is expressed in CNF.

(2) *Evaluation of the atoms with respect to the symmetrical threshold semantics*
After the query q is transformed into normal form, we evaluate all atoms according to the symmetrical threshold semantics by means of the function g^1_{2t}:

- For t_5 :

$$\{RSV_1^{5,1} = (VH, -.2), RSV_2^{5,1} = (H, .4),$$
$$RSV_5^{5,1} = (H, -.4), RSV_6^{5,1} = (VH, .2), RSV_7^{5,1} = (VH, .2)\}$$

- For t_6 :

$$\{RSV_1^{6,1} = (H, -.1), RSV_2^{6,1} = (L, .3), RSV_4^{6,1} = (H, -.5),$$
$$RSV_6^{6,1} = (L, -.46), RSV_7^{6,1} = (VH, .42)\}$$

- For t_7 :

$$\{RSV_1^{7,1} = (TO, 0), RSV_2^{7,1} = (VH, .1), RSV_4^{7,1} = (TO, 0),$$
$$RSV_6^{7,1} = (VH, -.3), RSV_7^{7,1} = (VH, -.3)\}$$

where, for example the $RSV_2^{7,1}$ is calculated as

$$RSV_2^{7,1} = g^1_{2t}(d_2, t_7, H) = \Delta(\Delta^{-1}(H, 0) + \frac{\Delta^{-1}(EH, .2) - \Delta^{-1}(H, 0)}{2}) = (VH, .1),$$

(with $k = 2$), given that the condition $(s_b, 0) \geq (s_{\frac{T}{2}}) \wedge (s_b, 0) \leq (s_a, \alpha_a) < (s_T, 0)$ is true.

If we apply the ordinal fuzzy linguistic IRS model,[1] that is, the linguistic matching function g^1, then we obtain the following linguistic relevance degrees:

- For t_5 :

$$\{RSV_1^{5,1} = VH, RSV_2^{5,1} = H, RSV_5^{5,1} = H,$$
$$RSV_6^{5,1} = VH, RSV_7^{5,1} = VH\}$$

- For t_6 :

$$\{RSV_1^{6,1} = H, RSV_2^{6,1} = L, RSV_4^{6,1} = H, \\ RSV_6^{6,1} = L, RSV_7^{6,1} = VH\}$$

- For t_7 :

$$\{RSV_1^{7,1} = TO, RSV_2^{7,1} = VH, RSV_4^{7,1} = TO, \\ RSV_6^{7,1} = VH, RSV_7^{7,1} = VH\}$$

In this step, it is easy to observe the effects of the use of 2-tuple linguistic representation, i.e., 2-tuple linguistic relevance results are richer than ordinal linguistic ones.

(3) *Evaluation of atoms with respect to the quantitative semantics*
The results of the evaluation of atoms of q according to the quantitative semantics modelled by g_{2t}^2 are the following:

- For t_5 :

$$\{RSV_6^{5,1,2} = (VH, .2)\}$$

- For t_6 :

$$\{RSV_1^{6,1,2} = (H, -.1), RSV_7^{6,1,2} = (VH, .42)\}$$

- For t_7 :

$$\{RSV_1^{7,1,2} = (TO, 0), RSV_4^{7,1,2} = (TO, 0)\}$$

where, for example, the $RSV_1^{7,1,2} = g_{2t}^2(d_2, RSV_1^{7,1}, c_7^2)$ is calculated as follows:
$K = \sharp \text{Supp}(M_7) = 5$, given that $Supp(M_7) = \{d_1, d_2, d_4, d_6, d_7\}$,
when $K = 2$ then the condition $(c_7^2, 0) = (L, 0) \geq (VL, .28) = S^2$ is true
therefore, we obtain $\beta^S = \{d_1, d_4\}$,
so, $RSV_1^{7,1,2} = g_{2t}^2(d_2, RSV_1^{7,1}, c_7^2) = RSV_1^{7,1} = (TO, 0)$, because $d_1 \in \beta^S$.
On the other hand, in the case of the ordinal fuzzy linguistic IRS model[1] using the matching function g^2 the results obtained are

- For t_5 :

$$\{RSV_6^{5,1,2} = VH\}$$

- For t_6 :

$$\{RSV_1^{6,1,2} = H, RSV_7^{6,1,2} = VH\}$$

- For t_7 :

$$\{RSV_1^{7,1,2} = TO, RSV_4^{7,1,2} = TO\}$$

We should note that the quantitative semantics decreases the number of documents associated to be considered in each query term. Really, the 2-tuple linguistic representation does not affect anything in this step of evaluation.

(4) *Evaluation of subexpressions and modelling the relative importance semantics*
The query q' has two subexpressions and both have two atoms, $q'_1 = (t_5, VH, VL, VH) \vee (t_7, H, L, H)$ and $q'_2 = (t_6, L, L, VL) \vee (t_7, H, L, H)$. Each subexpression is in disjunctive form, and thus, we must use a 2-tuple LOWA operator ϕ_{2t}^2 with orness measure $orness(W) > 0.5$ (for example, with ($W =$

$[0.8, 0.2]))$ together with the transformation function $MIN(Weight$, 2-tuple linguistic value) to evaluate them. Then, the results of evaluation applying the relative importance semantics are:

- For q_1' :

$$\{RSV_1^1 = (M, 0), RSV_4^1 = (M, 0), RSV_6^1 = (H, -.2)\}$$

- For q_2' :

$$\{RSV_1^2 = (M, .4), RSV_4^2 = (M, 0), RSV_7^2 = (VL, -.4)\}$$

where RSV_j^v is the result of evaluating the document d_j with respect to the subexpression $q_v', v \in \{1, 2\}$.
For example RSV_1^2 is calculated as

$$RSV_1^2 = \phi_{2t}^2(MIN_{2t}((c_6^3, 0), RSV_1^{6,1,2}), MIN_{2t}((c_7^3, 0), RSV_1^{7,1,2}))$$

That is,

$$RSV_1^2 = \phi_{2t}^2(MIN_{2t}((VL, 0), (H, -.1)), MIN_{2t}((H, 0), (TO, 0))) =$$

$$= \phi_{2t}^2((VL, 0), (H, 0)) \Rightarrow$$

$$RSV_1^2 = \phi_{2t}^2((H, 0), (VL, 0)) = \Delta(\Delta^{-1}(H, 0) \cdot 0.8 + \Delta^{-1}(VL, 0) \cdot 0.2) =$$

$$= \Delta(5 \cdot 0.8 + 2 \cdot 0.2) = \Delta(4.4) = (M, .4).$$

In the case of the ordinal fuzzy linguistic IRS model,[1] that is, using the linguistic t-conorm MAX together with the transformation function $MIN(Weight, value)$ to evaluate the disjunctive subexpressions we obtain the following:

- For q_1' :

$$\{RSV_1^1 = H, RSV_4^1 = H, RSV_6^1 = VH\}$$

- For q_2' :

$$\{RSV_1^2 = H, RSV_4^2 = H, RSV_7^2 = VL\}.$$

We should point out that in general the 2-tuple LOWA operator decreases the inclusive effect of the linguistic t-conorm MAX to calculate the linguistic relevance degrees.

(5) *Evaluation of the whole query* We obtain the evaluation of the whole query using a 2-tuple LOWA operator ϕ_{2t}^1 with $orness(W) < 0.5$ (e.g. with $(W = [0.2, 0.8])$).

$$\{RSV_1 = (M, .08), RSV_4 = (M, 0), RSV_6 = (EL, -.08), RSV_7 = (N, .32)\}.$$

The best retrieved documents is d_1, which is calculated as:

$$RSV_1 = \phi_{2t}^1(RSV_1^2, RSV_1^1) = \phi_{2t}^1((M, .4), (M, 0)) = \Delta(\Delta^{-1}(M, .4) \cdot 0.2 +$$

$$\Delta^{-1}(M, 0) \cdot 0.8) = \Delta(4.08) = (M, .08).$$

In the case of the ordinal fuzzy linguistic IRS model[1] the final result achieved by using the linguistic t-norm MIN is

$$\{RSV_1 = H, RSV_4 = H\}.$$

In this case, we achieve two best documents, d_1 and d_4, without possibility to distinguish between them. If we shall search information on the Web applying an ordinal fuzzy linguistic approach we would obtain many documents with the same linguistic relevance degree and we would not be able to distinguish between relevant and non-relevant documents. Therefore, our 2-tuple fuzzy linguistic IRS allows to work with a finest relevance representation.
In Figure 4 we show graphically the whole example of operation of this 2-tuple fuzzy linguistic IRS model.

3.3. *Evaluation with respect to the ordinal fuzzy IRS: Advantages and drawbacks*

In this subsection, we compare our 2-tuple fuzzy linguistic IRS (called SRI_{2t}) with respect to the ordinal fuzzy linguistic IRS defined in[1] (called SRI_o).

To do that, we have worked with the well known CACM documentary base to test the performance of our proposal. The 3204 documents of CACM have been automatically indexed by first extracting the non-stop words, and then using the normalized IDF scheme to generate the term weights in the document. CACM has got 64 predefined queries, which have been extended to Boolean ordinal weighted queries by weighting its terms with three ordinal linguistic values using the terms set of Example 2 and the five rules given in the Query Component defined in Section 2.2. For example, query number 8 is compound by six terms: *address*, *operat*, *schem*, *resourc*, *network*, *system* and it is extended to the Boolean ordinal weighted query as follows:

$$\langle address, L, VH, H\rangle \; AND \; \langle operat, M, TO, VH\rangle \; AND \; \langle schem, L, VH, VH\rangle \; AND$$

$$\langle resourc, VL, TO, TO\rangle \; AND \; \langle network, L, M, H\rangle \; AND \; \langle system, M, TO, H\rangle.$$

From this experiment, we can conclude the following:

- *With respect to the information representation*: The system SRI_{2t} allows us to distinguish the better documents. For example, we can observe this if we compare the results of SRI_o and SRI_{2t} for the above query which are shown in Tables 1 and 2, respectively. Both SRI_o and SRI_{2t} achieve same documents set, but SRI_{2t} is able to obtain a best ranking of them.
- *With respect to the evaluation operators:* The use of the 2-tuple LOWA operator to model the connectives AND and OR incorporates more flexibility in the computation of the results. For example, in Table 2 we have used the operator LOWA_{2t} with an $orness(W) = 0$, that is, we have used an operator equivalent

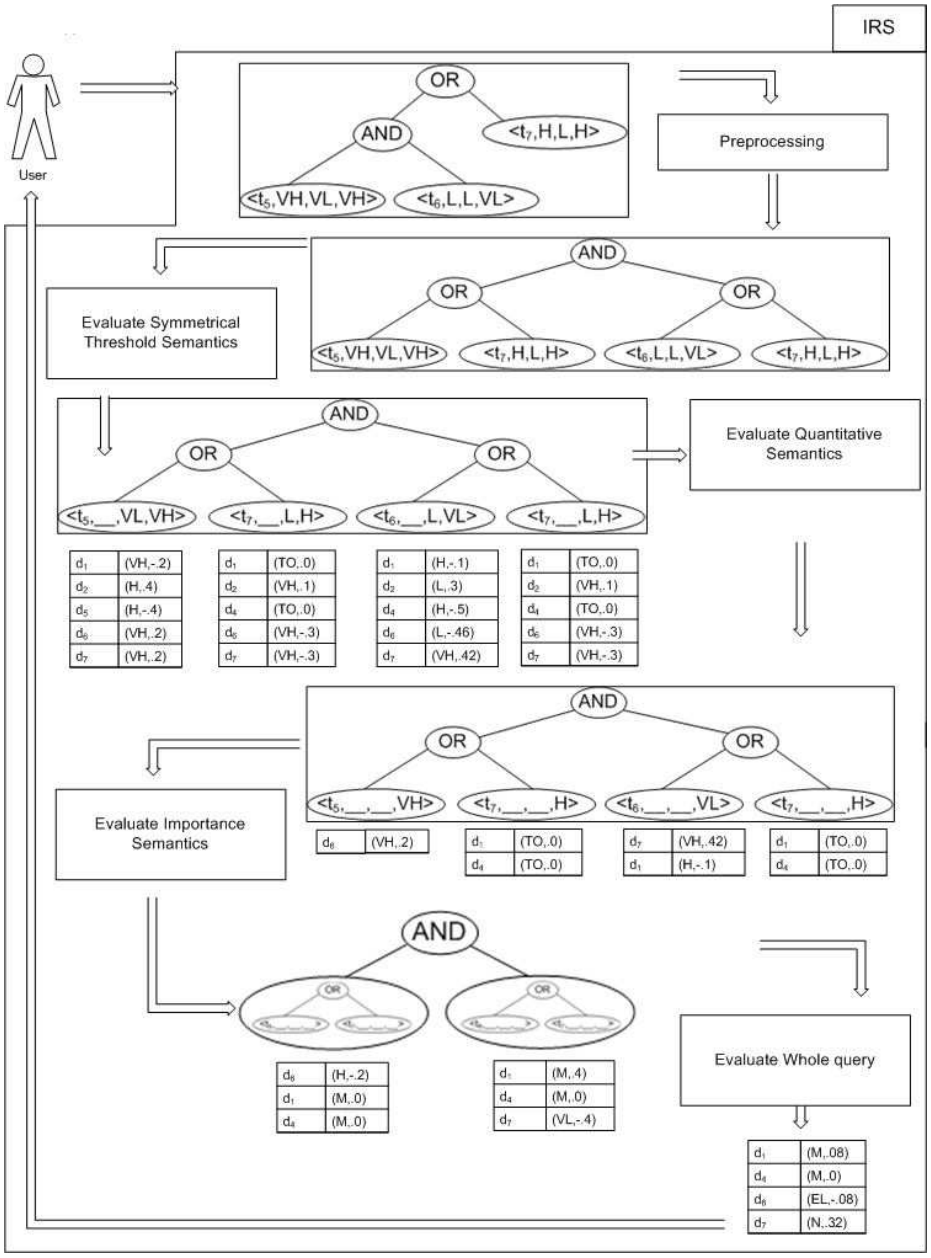

Fig. 4. Information retrieval process.

to the t-norm MIN used in SRI_o. In such a way, we obtain the same documents. This happens always, that is, SRI_{2t} retrieves at least the same documents as SRI_o does. It is possible that SRI_{2t} retrieves more documents by softening the

Table 1. Relevance results of SRI_o.

Rank	**ID Doc**	**RSV**
1#	2967	VH
...	2895	VH
...	2785	VH
...	2060	VH
...	1747	VH
...	1471	VH
...	1262	VH

Table 2. Relevance results of SRI_{2t}.

Rank	**ID Doc**	**RSV**
1#	2967	(VH,0.09)
2#	2060	(VH,0.08)
3#	1747	(VH,0.01)
4#	1471	(VH,-0.06)
5#	2785	(VH,-0.29)
6#	2895	(VH,-0.36)
7#	1262	(VH,-0.44)

Table 3. Relevance results of SRI_{2t} with $orness = 0.2$.

Rank	**ID Doc**	**RSV**
1#	2967	(VH,0.23)
2#	2060	(VH,0.19)
3#	1747	(VH,0.16)
4#	1471	(VH,0.07)
5#	2785	(VH,-0.00)
6#	1262	(VH,-0.03)
7#	2895	(VH,-0.05)
8#	2002	(M,0.24)
9#	1315	(M,0.23)
10#	2922	(M,0.23)
11#	2396	(M,0.22)
12#	3077	(M,0.22)
13#	2106	(M,0.22)

restrictive behavior of the AND connective. For example, in Table 3 using a LOWA_{2t} with $orness = 0.2$) we have more documents, some relevant (8, 9, 10) and others not (11, 12, 13).

- *With respect to the precision and recall:* Both SRI_o and SRI_{2t} in similar conditions, that is, using a LOWA_{2t} with $orness = 0$ and a LOWA_{2t} with $orness = 1$ to model the connectives AND and OR, respectively, present similar precision and recall indexes. However, when we change the *orness* the precision and

recall measures change also. For example, if we use a LOWA_{2t} with $orness = 0$ to model the connective AND and then we use a LOWA_{2t} with $orness = 0.2$, we observe that the precision measure is decreased and the recall is increased. Inverse behaviour is observed in the case of the connective OR.

Finally, we should analyze the main advantages and drawbacks of SRI_{2t} with respect to SRI_o.

Advantages.
- Firstly, it is obvious the advantage of the use of the 2-tuple fuzzy linguistic representation model in SRI_{2t}, given that if we use an ordinal linguistic representation it is impossible to distinguish the relevance difference between some documents.
- Secondly, also it is obvious that the 2-tuple fuzzy linguistic representation model avoids the loss of information in the computation process of relevance degrees.
- Thirdly, with the 2-tuple fuzzy linguistic representation model in SRI_{2t} the complexity of some matching functions is simplified, as it is the case of the quantitative semantics.
- Fourthly, the new linguistic operator proposed to model the logical connectives AND and OR, the 2-tuple LOWA operator, incorporates more flexibility in the computation of the results.
- We should point out that this new linguistic IRS model improves in general the performance of SRI_o with a minimum cost and without to affect negatively to the IRS-user interaction, given that the query language is the same and the relevance degrees continue being expressed in a linguistic way.

Drawbacks. We observe the similar drawbacks that affect to the IRS model proposed in.[1] Mainly two:
- With the query subsystem user can express a large number of requirements, but he must decide what and how many semantics must be considered for formulating his/her information needs, the system supports all the possibilities. Therefore, it is necessary the design of an adequate user interface that could help users to make better use of the expression possibilities of the weighted query language.
- To define tools that could allow users to control the aggregations in the evaluation process, i.e., involving the concept of users' relevance in the level of Boolean logical connectives.

4. Conclusions

In this paper, we have presented a new linguistic IRS model based on the 2-tuple fuzzy linguistic approach. Such a linguistic approach allows to avoid the problems of loss of precision and lack of information detected in the ordinal fuzzy linguistic IRS activity, and consequently, it improves its performance. This improvement is achieved because the evaluation of the relevance of documents is not expressed

only by means of a single label, but also it has associated a translation value that stores an information, which in the ordinal case is discarded. Additionally, we have incorporated a new operator, the 2-tuple LOWA operator, which allows to soften the modelling of the Boolean logical connectives AND and OR, and in such a way, to contribute to improve the retrieval results.

In the future, we will study mechanisms to improve the performance of this linguistic IRS model. We think that a possible solution could consist to incorporate more information in the system about the concept of relevance that users present. For example, this could be achieved by defining the linguistic matching functions and the aggregation operators depending on the user's parameters.

References

1. E. Herrera-Viedma, Modelling 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):460–475, 2001.
2. E. Herrera-Viedma, An information retrieval system with ordinal linguistic weighted queries based on two weighting elements, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9:77–88, 2001.
3. A. Bookstein, Fuzzy request: An approach to weighted boolean searches, *Journal of the American Society for Information Science and Technology*, 31(4):240–247, 1980.
4. G. Bordogna, P. Carrara, and G. Pasi, Query term weights as constraints in fuzzy information retrieval, *Information Processing and Management*, 27(1):15–26, 1991.
5. G. Bordogna and G. Pasi, Linguistic aggregation operators of selecction criteria in fuzzy information retrieval, *International Journal of Intelligent Systems*, 10:233–248, 1995.
6. D. Buell and D. H. Kraft, A model for a weighted retrieval system, *Journal of the American Society for Information Science*, 32:211–216, 1981.
7. D. Buell and D. H. Kraft, Threshold values and boolean retrieval systems, *Information Processing and Management*, 17:127–136, 1981.
8. C. S. Cater and D. H. Kraft, A generalization and clarification of the Waller-Kraft wish-list, *Information Processing and Management*, 25:15–25, 1989.
9. D. H. Kraft and D. A. Buell, Fuzzy sets and generalized boolean retrieval systems, *International Journal of Man-Machine Studies*, 19:45–56, 1983.
10. W. G. Waller and D. H. Kraft, A mathematical model of a weighted boolean retrieval system, *Information Processing and Management*, 15:235–245, 1979.
11. R. R. Yager, A hierarchical document retrieval language, *Information Retrieval*, 3:357–377, 2000.
12. R. R. Yager, A note on weighted queries in information retrieval systems, *Journal of the American Society of Information Sciences*, 38:23–24, 1987.
13. 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(2):70–82, 1993.
14. G. Bordogna and G. Pasi, An ordinal information retrieval model, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9:63–76, 2001.
15. E. Herrera-Viedma, O. Cordón, M. Luque, A. G. López-Herrera, and A. M. Muñoz, A model of fuzzy linguistic IRS based on multi-granular linguistic information, *International Journal of Approximate Reasoning*, 34:221–239, 2003.
16. D. H. Kraft, G. Bordogna, and G. Pasi, An extended fuzzy linguistic approach to generalize boolean information retrieval, *Information Sciences*, 2:119–134, 1994.

17. L. A. Zadeh, The concept of a linguistic variable and its applications to approximate reasoning, Part I. *Information Sciences*, 8:199–249, 1975.
18. L. A. Zadeh, The concept of a linguistic variable and its applications to approximate reasoning, Part II. *Information Sciences*, 8:301–357, 1975.
19. L. A. Zadeh, The concept of a linguistic variable and its applications to approximate reasoning, Part III. *Information Sciences*, 9:43–80, 1975.
20. F. Herrera and E. Herrera-Viedma, Aggregation operators for linguistic weighted information, *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, 27:646–656, 1997.
21. F. Herrera, E. Herrera-Viedma, and J. L. Verdegay, Direct approach processes in group decision making using linguistic OWA operators, *Fuzzy Sets and Systems*, 79:175–190, 1996.
22. R. R. Yager, An approach to ordinal decision making, *International Journal of Approximate Reasoning*, 12:237–261, 1995.
23. E. Herrera-Viedma, G. Pasi, A. G. López-Herrera, and C. Porcel, Evaluating the information quality of Web sites: A qualitative methodology based on fuzzy computing with words, *Journal of the American Society for Information Science and Technology*, 57(4):538–549, 2006.
24. F. Herrera, E. Herrera-Viedma, and J. L. Verdegay, A linguistic decision process in group decision making. *Group Decision and Negotiation*, 5:165–176, 1996.
25. F. Herrera and L. Martínez, A 2-tuple fuzzy linguistic representation model for computing with words, *IEEE Transactions on Fuzzy Systems*, 8(6):746–752, 2000.
26. C. Carlsson and R. Fuller, Benchmarking and linguistic importance weighted aggregations, *Fuzzy Sets and Systems*, 114:35–41, 2000.
27. B. Arfi, Fuzzy decision making in politics: A linguistic fuzzy-set approach (LFSA), *Political Analysis*, 13(1):23–56, 2005.
28. J. Domingo-Ferrer and V. Torra, Median-based aggregation operators for prototype construction in ordinal scales, *International Journal of Intelligent Systems*, 18:633–655, 2003.
29. E. Herrera-Viedma, F. Herrera, L. Martínez, J. C. Herrera, and A. G. López-Herrera, Incorporating filtering techinques in a fuzzy linguistic multi-agent model for gathering of information on the Web, *Fuzzy Sets and Systems*, 148(1):61–83, 2004.
30. E. Herrera-Viedma and E. Peis, Evaluating the informative quality of documents in SGML-format using fuzzy linguistic techniques based on computing with words, *Information Processing and Management*, 39(2):195–213, 2003.
31. J. Kacprzyk and S. Zadrozny, Computing with words for text processing: an approach to the text categorization, *Information Sciences*, 176(4):415–437, 2006.
32. V. Torra, Aggregation of linguistic labels when semantics is based on antonyms, *International Journal of Intelligent Systems*, 16:513–524, 2001.
33. Z. Xu, Deviation measures of linguistic preference relations in group decision making, *Omega International Journal of Management Science*, 33(3):249–254, 2005.
34. P. P. Bonissone and K. S. Decker, *Uncertainty in Artificial Intelligence*, chapter Selecting Uncertainty Calculi and Granularity: An Experiment in Trading-off Precision and Complexity, pages 217–247, L. H. Kanal and J. F. Lemmer, Eds. (North–Holland), 1986.
35. R. R. Yager, On ordered weighted averaging aggregation operators in multicriteria decision making, *IEEE Transactions on Systems, Man, and Cybernetics*, 18:183–190, 1988.
36. M. Delgado, J. L. Verdegay, and M. A. Vila, On aggregation operations of linguistic labels, *International Journal of Intelligent Systems*, 8:351–370, 1993.