# Improving the performance of ordinal fuzzy linguistic IRSs*

**E. Herrera-Viedma** **O. Cordón** **M. Luque**
Dept. of Computer Science and A.I, Dept. of Computer Science and N.A.,
University of Granada, Granada (Spain) University of Córdoba, Córdoba (Spain)
viedma@decsai.ugr.es ocordon@decsai.ugr.es mluque@uco.es

## Abstract

Information Retrieval Systems (IRSs) based on an ordinal fuzzy linguistic approach present some problems of loss of precision and information when working with discrete linguistic expression domains or when applying approximation operations in the symbolic aggregation methods.

In this paper, we present an IRS based on a 2-tuple fuzzy linguistic approach which allows us to overcome the problems of ordinal fuzzy linguistic IRSs and improve the performance.

**KeyWords:** Information Retrieval, Fuzzy Linguistic Modelling.

## 1 Introduction

Information Retrieval involves the development of computer systems for the storage and retrieval of (predominantly) textual information (documents). The use of linguistic variables [15] to represent the input and output information in the retrieval process of Information Retrieval Systems (IRSs) considerably improves the IRS-user interaction. In the literature we can find different models of linguistic IRSs based on a *fuzzy linguistic approach* [1, 2, 3, 6, 7, 8, 9].

Linguistic IRS Models proposed in [3, 6, 7, 8] are based on an ordinal fuzzy linguistic approach [4], and they are affected by the two characteristic problems of ordinal fuzzy linguistic modelling [5]:

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

The main aim of the paper is to present a model of a linguistic IRS based on a 2-tuple fuzzy linguistic approach [5]. The 2-tuple fuzzy linguistic modelling solves the problems of ordinal one, and therefore, allows us to improve the performance of ordinal fuzzy linguistic IRSs.

This contribution is set out as follows. The 2-tuple fuzzy linguistic approach is presented in Section 2. The new model of linguistic IRS is defined in Section 3. Finally, Section 4 includes our conclusions.

## 2 The 2-tuple fuzzy linguistic approach

The 2-tuple fuzzy linguistic approach is a special kind of fuzzy linguistic approach that was introduced in [5] to overcome the problems of the ordinal one [7].

Let $S = \{s_0, ..., s_T\}$ be a linguistic term set with odd cardinality ($T+1$ is the cardinality of $S$ and usually is equal to 7 or 9[1]), where the mid term represents an assessment of approximately 0.5 and with the rest of the terms being placed symmetrically around it.

**Definition 1.** [5] Let $\beta$ be the result of an aggregation of the indexes of a set of labels assessed in $S$, i.e., the result of a symbolic aggregation operation, $\beta \in [0, T]$ and $\beta \notin \{0, ..., T\}$,. Let $i = round(\beta)$ and $\alpha = \beta - i$ be

*This research has been supported by CICYT under project TIC2003-07977

[1] Several studies have demonstrated that people can not handle more than 7+/-2 levels of quantification [11].

two values, such that, $i \in \{0, ..., \mathcal{T}\}$ and $\alpha \in [-.5, .5)$ then $\alpha$ is called a *Symbolic Translation*.

The 2-tuple fuzzy linguistic approach is developed from the concept of symbolic translation by representing the linguistic information equivalent to $\beta$ by means of 2-tuples $(s_i, \alpha_i)$, $s_i \in S$ and $\alpha_i \in [-.5, .5)$. This model defines a set of transformation functions between numeric values and 2-tuples.

**Definition 2.** [5] Let $S$ be a linguistic term set and $\beta \in [0, \mathcal{T}]$, then the 2-tuple that expresses the equivalent information to $\beta$ is obtained with the following function: $\Delta : [0, \mathcal{T}] \longrightarrow S \times [-0.5, 0.5)$,

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

For all $\Delta$ there exists $\Delta^{-1}$, defined as $\Delta^{-1}(s_i, \alpha) = i + \alpha$. Obviously, the conversion of a linguistic term into a linguistic 2-tuple consists of adding a symbolic translation value of 0: $s_i \in S \Longrightarrow (s_i, 0)$.

The 2-tuple linguistic computational model operates with the 2-tuples without loss of information and is based on the following operations [5]:

***1. Negation operator of 2-tuples:***

$$Neg((s_i, \alpha)) = \Delta(\mathcal{T} - (\Delta^{-1}(s_i, \alpha))).$$

***2. Comparison of 2-tuples:*** The comparison of linguistic 2-tuples is carried out according to an ordinary lexicographic order. Let $(s_k, \alpha_1)$ and $(s_l, \alpha_2)$ be two 2-tuples, with each one representing a counting of information:

- If $k < l$ then $(s_k, \alpha_1)$ is smaller than $(s_l, \alpha_2)$
- If $k = l$ then
  1. if $\alpha_1 = \alpha_2$ then $(s_k, \alpha_1)$ and $(s_l, \alpha_2)$ represent the same information,
  2. if $\alpha_1 < \alpha_2$ then $(s_k, \alpha_1)$ is smaller than $(s_l, \alpha_2)$,
  3. if $\alpha_1 > \alpha_2$ then $(s_k, \alpha_1)$ is bigger than $(s_l, \alpha_2)$.

***3. Aggregation of 2-tuples:*** Using $\Delta$ and $\Delta^{-1}$ any numerical aggregation operator can be easily extended for dealing with linguistic 2-tuples.

**Definition 3.** [14] Let $A = \{a_1, \ldots, a_m\}, a_k \in [0, 1]$ be a set of assessments to be aggregated, then the Ordered Weighted Averaging (OWA) operator, $\phi$, is defined as

$$\phi(a_1, \ldots, a_m) = W \cdot B^T$$

where $W = [w_1, \ldots, w_m]$, is a weighting vector, such that $w_i \in [0, 1]$ and $\Sigma_i w_i = 1$; and $B = [b_1, \ldots, b_m]$ is a vector associated to $A$, such that, $B = \sigma(A) = \{a_{\sigma(1)}, \ldots, a_{\sigma(m)}\}$, where $a_{\sigma(j)} \leq a_{\sigma(i)} \ \forall \ i \leq j$, with $\sigma$ being a permutation over the set of labels $A$.

A 2-tuple linguistic definition of $\phi$ would be as follows:

**Definition 4.** Let $A = \{(a_1, \alpha_1), \ldots, (a_m, \alpha_m)\}$ be a set of assessments in the 2-tuple linguistic domain, then the 2-tuple linguistic OWA operator, $\phi_{2t}$ is defined as:

$$\phi_{2t}((a_1, \alpha_1), \ldots, (a_m, \alpha_m)) = \Delta(W \cdot B^T)$$

$$B = \sigma(A) = \{\Delta^{-1}(a_1, \alpha_1))_{\sigma(1)}, \ldots, (\Delta^{-1}(a_m, \alpha_m))_{\sigma(m)}\}.$$

## 3 A 2-tuple fuzzy linguistic IRS model

In this Section, we present a fuzzy linguistic IRS model based on a 2-tuple fuzzy linguistic approach which overcomes the problems of loss of precision and information of the ordinal fuzzy linguistic IRS models. The main properties of this model linguistic IRS model are: i)users can express their information needs by means of multi-weighted linguistic Boolean queries using different semantics, even, simultaneously, and ii) the Boolean connectives are modelled in a flexible way by means of soft computing operator, 2-tuple linguistic OWA operator.

### 3.1 Multi-weighted linguistic Boolean queries

We consider a set of documents D= $\{d_1, \ldots, d_m\}$ represented by means of index terms T= $\{t_1, \ldots, t_l\}$, which describe the subject content of the documents. A numeric indexing function $F$ : DxT $\rightarrow [0, 1]$ is defined, called *index term weighting*. $F$ maps a given document $d_j$ and a given index term $t_i$ to a numeric weight between 0 and 1. Thus, $F(d_j, t_i)$ is a numerical weight that represents the degree of significance of $t_i$ in $d_j$. $F(d_j, t_i) = 0$ implies that the document $d_j$ is not at all about the concept(s) represented by the 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 $t_i$.

To retrieve documents from D users can use multi-weighted linguistic Boolean queries as in [6, 7, 8]. With such queries a information need is expressed as a combination of the index terms which are connected by the logical operators AND ($\wedge$), OR ($\vee$), and NOT ($\neg$) and can be weighted with three ordinal linguistic values [2] taken from a label set $S$ associated to three different semantics as in [7]: symmetrical threshold semantics, relative importance semantics, quantitative semantics.

[2]The weights are defined as ordinal values but they are transformed to 2-tuple values, adding a symbolic translation value of 0, in order to process the query.

As in [2] we use the linguistic variable *Importance* to express the linguistic weights, but defining it with an ordinal fuzzy linguistic approach [7]. Thus, we consider a set of labels $S$ to express the query weights. Then, we define a multi-weighted linguistic 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, $\mathrm{Tx}S^3$; $t_i \in \mathrm{T}$, and $c_i^1,\ c_i^2,\ c_i^3$ are ordinal values of the linguistic variable *Importance*, modelling the symmetrical threshold semantics, the quantitative semantics, and the importance semantics, respectively. Accordingly, the set Q of the legitimate queries is defined by the following syntactic rules:

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

### 3.2 Evaluating multi-weighted linguistic Boolean queries

The evaluation of a multi-weighted linguistic Boolean query is carried out by means of a constructive bottom-up process based on the *criterion of separability* [12] and at the same time as supporting all the semantics of query weights considered. The evaluation of a query is developed in the five subsequent steps:

1.- Preprocessing of the query.

The user query is preprocessed to put it into either conjunctive normal form (CNF) or disjunctive normal form (DNF), in such a way that every Boolean subexpression must have more than two atoms. Weighted single-term queries are kept in their original forms.

2.- Evaluation of atoms with respect to the symmetrical threshold semantics.

According to a symmetrical threshold semantics, a user may search for documents with a minimally acceptable presence of one term in their representations, or documents with a maximally acceptable presence of one term in their representations [7, 6]. Given a request $< t_i, c_i^1, c_i^2, c_i^3 >$, this means that the query weights that imply the presence of a term in a document $c_i^1 \geq s_{\mathcal{T}/2}$ (e.g. *High, Very High*) must be treated differently to the query weights that imply the absence of one term in a document $c_i^1 < s_{\mathcal{T}/2}$ (e.g. *Low, Very Low*). Then, if $c_i^1 \geq s_{\mathcal{T}/2}$, the request $< t_i, c_i^1, c_i^2, c_i^3 >$ is synonymous with the request $< t_i, at\ least\ c_i^1, c_i^2, c_i^3 >$, which expresses the fact that the desired documents are those having $F$ values as high as possible; and if $c_i^1 < s_{\mathcal{T}/2}$, the former request is synonymous with the request $< t_i, at\ most\ c_i^1, c_i^2, c_i^3 >$, which expresses the fact that the desired documents are those having $F$ values as low as possible. This interpretation is defined by means of a parameterized 2-tuple linguistic matching function $g^1 : D\times$ T $\times(S \times [-.5, .5)) \rightarrow (S \times [-.5, .5))$. Given an atom $< t_i, c_i^1, c_i^2, c_i^3 >$ and a document $d_j \in D$, $g^1$ obtains the linguistic RSV of $d_j$, called $RSV_j^{i,1}$, by measuring how well the index term weight $F(d_j, t_i)$ satisfies the request expressed by the linguistic weight $c_i^1$ according to the following expression:

$$RSV_j^{i,1} = g^1(d_j, t_i, (c_i^1, 0)) =$$

$$\begin{cases} \Delta(0) & \text{if } (s_b,0) \geq (s_{\mathcal{T}/2},0) \text{ and } (s_a,0) = (s_0,0) \\ \Delta(i_1) & \text{if } (s_b,0) \geq (s_{\mathcal{T}/2},0) \text{ and } (s_0,0) < (s_a,0) < (s_b,0) \\ \Delta(i_2) & \text{if } (s_b,0) \geq (s_{\mathcal{T}/2},0) \text{ and } (s_b,0) \leq (s_a,0) < (s_{\mathcal{T}},0) \\ \Delta(\mathcal{T}) & \text{if } (s_b,0) \geq (s_{\mathcal{T}/2},0) \text{ and } (s_a,0) = (s_{\mathcal{T}},0) \\ \Delta(\mathcal{T}) & \text{if } (s_b,0) < (s_{\mathcal{T}/2},0) \text{ and } (s_a,0) = (s_0,0) \\ Neg(\Delta(i_1)) & \text{if } (s_b,0) < (s_{\mathcal{T}/2},0) \text{ and } (s_0,0) < (s_a,0) \leq (s_b,0) \\ Neg(\Delta(i_2)) & \text{if } (s_b,0) < (s_{\mathcal{T}/2},0) \text{ and } (s_b,0) < (s_a,0) < (s_{\mathcal{T}},0) \\ \Delta(0) & \text{if } (s_b,0) < (s_{\mathcal{T}/2},0) \text{ and } (s_a,0) = (s_{\mathcal{T}},0) \end{cases}$$

such that

$$\begin{array}{l} i_1 = Max\{0, (b - \frac{(b-a)}{\mathcal{K}})\} \\ i_2 = Min\{\mathcal{T}, (b + \frac{(a-b)}{\mathcal{K}})\} \end{array} \quad \mathcal{K} \in \{1, 2, 3, ..., b\}.$$

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

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

3.- Evaluation of atoms with respect to the quantitative semantics.

In this step, documents go on being 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 [7]: when a user establishes a certain number 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 $RSV_j^{i,1} \in (S\times[-.5, .5))$ 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 $g^2$, which is defined between the $RSV_j^{i,1}$ and the linguistic quantitative weight $c_i^2 \in S$. Then, 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 $RSV_j^{i,1,2} \in (S \times [-.5, .5))$, is obtained by means of the linguistic matching function $g^2 : D \times (S \times [-.5, .5)) \times S \rightarrow (S \times [-.5, .5))$ as follows $RSV_j^{i,1,2} =$

$$g^2(RSV_j^{i,1}, c_i^2, d_j) = \begin{cases} (s_0, 0) & \text{if } d_j \notin \mathcal{B}^{\mathcal{S}} \\ RSV_j^{i,1} & \text{if } d_j \in \mathcal{B}^{\mathcal{S}} \end{cases} \quad \text{where}$$

$\mathcal{B}^{\mathcal{S}}$ is the set of documents such that $\mathcal{B}^S \subseteq Supp(\mathcal{M})$ where $\mathcal{M} = \{(d_1, RSV_1^{i,1}), \ldots, (d_m, RSV_m^{i,1})\}$, is a fuzzy subset of documents obtained according to the following algorithm:

1. $K = \#Supp(\mathcal{M})$
2. REPEAT
   $M^K = \{s_q \in S : \mu_{s_q}(K/m) = Sup_v\{\mu_{s_v}(K/m)\}\}$.
   $s^K = Sup_q\{s_q \in M^K\}$.
   $K = K - 1$.
3. UNTIL $((c_i^2 \in M^{K+1})$ OR $(c_i^2 \geq s^{K+1}))$.
4. $\mathcal{B}^{\mathcal{S}} = \{d_{\sigma(1)}, \ldots, d_{\sigma(K+1)}\}$, 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 consists of reducing the number of documents to be considered in the evaluation of $t_i$ in the later steps.

4.- Evaluation of subexpressions and modelling of the relative importance semantics

We consider that the relative importance semantics in a single-term query has no meaning. Then, in this step we have to evaluate the relevance of documents with respect to the subexpressions of queries composed of two atomic components..

Given a subexpression $q_v$ with $\mathcal{I} \geq 2$ atoms, we know that each document $d_j$ presents a partial $RSV_j^{i,1,2} \in (S \times [-.5, .5))$ 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 $\{RSV_j^{i,1,2},\ i = 1, \ldots, \mathcal{I}\}$ weighted by means of the respective relative importance degrees $\{c_i^3 \in S,\ i = 1, \ldots, \mathcal{I}\}$.

In each subexpression $q_v$ we find that the atoms can be combined using the AND or OR Boolean connectives, depending on the normal form of the user query. The restrictions imposed by the relative importance weights must be applied in the aggregation operators used to model both connectives. These aggregation operators should guarantee that the more important the query terms, the more influential they are in the determination of the RSVs. To do so, these aggregation operators must carry out two activities [4]: i) the transformation of the weighted information under the importance degrees by means of a transformation function $h$; and ii) 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 [13], Yager discussed the effect of the importance degrees on the MAX (used to model the connective OR) and MIN (used to model the connective AND) 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. Similarity, in [10], Lee analizes the behavioral aspects of the fuzzy operators and address important issues to affect retrieval effectiveness.

Following the ideas shown above, we use the extended definition of OWA operators $\phi_{2t}^1$ (with orness(W)$\leq$ 0.5) and $\phi_{2t}^2$ (with orness(W)$>$ 0.5) to model the AND and OR connectives, respectively. [3] Hence, when $h = \phi_{2t}^1$, then $f = \max(Neg(weight), value)$, and when $h = \phi^2$, then $f = min(weight, value)$.

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

1. if $q_v$ is a conjunctive subexpression then $RSV_j^v = \phi_{2t}^1(max(Neg(c_1^3, 0), RSV_j^{1,1,2}), \ldots, max(Neg(c_{\mathcal{I}}^3, 0), RSV_j^{\mathcal{I},1,2}))$, *and*
2. if $q_v$ is a disjunctive subexpression then $RSV_j^v = \phi_{2t}^2(min((c_1^3, 0), RSV_j^{1,1,2}), \ldots, min((c_{\mathcal{I}}^3, 0), RSV_j^{\mathcal{I},1,2}))$.

5.- Evaluation of the whole query.

In this step, the final evaluation of each document is achieved by combining their evaluations with respect

[3] In order to classify OWA operators in regards to their location between '"and" and "or" Yager [14] introduced a *orness measure*, associated with any vector W as follow: $orness(W) = \frac{1}{(m-1)} \sum_{i=1}^{m} (m - i) w_i$

to all the subexpressions using, again, the extended definition of OWA operators $\phi^1_{t2}$ and $\phi^2_{t2}$ to model the AND and OR connectives, respectively.

Then, given a document $d_j$, we evaluate its relevance with respect to a query $q$ as $RSV_j \in (S \times [-.5, 5))$ where $RSV_j = \phi^1_{2T}(RSV^1_j, \ldots, RSV^{\mathcal{V}}_j)$ if $q$ is in CNF, and $RSV_j = \phi^2_{2T}(RSV^1_j, \ldots, RSV^{\mathcal{V}}_j)$ if $q$ is in DNF, with $\mathcal{V}$ standing for the number of subexpressions in $q$.

**Remark 1:** *On the NOT Operator.* We should note that, if a query is in CNF or DNF form, 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 [7], the evaluation of document $d_j$ for a negated weighted atom $< \neg(t_i), c^1_i, c^2_i, c^3_i >$ is obtained from the negation of the index term weight $F(t_i, d_j)$. This means to calculate $g^1$ from the linguistic value $Label(1 - F(t_i, d_j))$.

### 3.3 Example of Application

In this subsection, we present an example of performance of the proposed IRS let us suppose a small database containing a set of seven documents $D = \{d_1, \ldots, d_7\}$, represented by means of a set of 10 index terms $T = \{t_1, \ldots, t_{10}\}$. Documents are indexed by means of an indexing function $F$, which assigns the following weights to each of them:

$$\begin{aligned}
d_1 &= 0.7/t_5 + 0.4/t_6 + 1/t_7 \\
d_2 &= 1/t_4 + 0.6/t_5 + 0.8/t_6 + 0.9/t_7 \\
d_3 &= 0.5/t_2 + 1/t_3 + 0.8/t_4 \\
d_4 &= 0.9/t_4 + 0.5/t_6 + 1/t_7 \\
d_5 &= 0.7/t_3 + 1/t_4 + 0.4/t_5 + 0.8/t_9 + 0.6/t_{10} \\
d_6 &= 1/t_5 + 0.99/t_6 + 0.8/t_7 \\
d_7 &= 0.8/t_5 + 0.02/t_6 + 0.8/t_7 + 0.9/t_8.
\end{aligned}$$

In the same way, let us suppose the following label set to express the values of the linguistic variable *Importance*.

$S = \{N = (0, 0, 0, 0), EL = (0.01, 0.02, 0.01, 0.05),$
$VL = (0.1, 0.18, 0.06, 0.05), L = (0.22, 0.36, 0.05, 0.06),$
$M = (0.41, 0.58, 0.09, 0.07), H = (0.63, 0.80, 0.05, 0.06),$
$VH = (0.78, 0.92, 0.06, 0.05), EH = (0.98, 0.99, 0.05, 0.01)$
$, T = (1, 1, 0, 0)\}$.

Finally, consider that a user formulates the following query $q = ((t_5, VH, VL, VH) \wedge (t_6, L, L, VL)) \vee (t_7, H, L, H)$. Its evaluation is as follows:

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 normal form with every subexpression having more than 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 atoms with respect to the symmetrical threshold semantics.

In this step we obtain the documents represented in a linguistic form using the translation function Label:

$$\begin{aligned}
d_1 &= H/t_5 + M/t_6 + T/t_7 \\
d_2 &= T/t_4 + M/t_5 + H/t_6 + VH/t_7 \\
d_3 &= M/t_2 + T/t_3 + H/t_4 \\
d_4 &= VH/t_4 + M/t_6 + T/t_7 \\
d_5 &= H/t_3 + T/t_4 + M/t_5 + H/t_9 + M/t_{10} \\
d_6 &= T/t_5 + EH/t_6 + H/t_7 \\
d_7 &= H/t_5 + EL/t_6 + H/t_7 + VH/t_8.
\end{aligned}$$

Let us set the sensitivity parameter $\mathcal{K} = 2$ which gives a large importance to the closeness between linguistic values in $g^1$. Then, the evaluations of atoms according to the symmetric threshold semantics modelled by $g^1$ are:
$\{RSV^{5,1}_1 = (VH, -.5), RSV^{5,1}_2 = (H, 0), RSV^{5,1}_5 = (H, 0)$
$, RSV^{5,1}_6 = (T, 0), RSV^{5,1}_7 = (VH, -.5)\}$
$\{RSV^{6,1}_1 = (H, -.5), RSV^{6,1}_2 = (M, 0), RSV^{6,1}_4 = (H, -.5)$
$, RSV^{6,1}_6 = (L, 0), RSV^{6,1}_7 = (VH, 0)\}$
$\{RSV^{7,1}_1 = (T, 0), RSV^{7,1}_2 = (VH, -.5), RSV^{7,1}_4 = (T, 0)$
$, RSV^{7,1}_6 = (H, 0), RSV^{7,1}_7 = (H, 0)\}$

3.- Evaluation of atoms with respect to the quantitative semantics.

The evaluation of the atoms of the query according to the quantitative semantics modeled by $g^2$ are:
$\{RSV^{5,1,2}_6 = (T, 0)\}$

$\{RSV^{6,1,2}_7 = (VH, 0), RSV^{6,1,2}_1 = (H, -.5)\}$

$\{RSV^{7,1,2}_1 = (T, 0), RSV^{7,1,2}_4 = (T, 0)\}$

We should note that the quantitative semantics decreases the number of documents associated to be considered in each query term.

4.- Evaluation of subexpressions and modelling the relative importance semantics.

The query $q'$ has two subexpressions and each of them presents 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 an OWA operator $\phi^2_{2t}$ with $orness(W) > 0.5$ (for example, with $(W = [0.8, 0.2])$).

$\{RSV^1_1 = (M, 0), RSV^1_4 = (M, 0), RSV^1_6 = (H, -.2)\}$

$\{RSV^2_1 = (M, .4), RSV^2_4 = (M, 0), RSV^2_7 = (VL, -.4).\}$

5.- Evaluation of the whole query.

We obtain the document evaluation with respect to the whole query using an OWA operator $\phi^1_{2t}$ with $orness(W) < 0.5$ (e.g. with $(W = [0.4, 0.6])$).

$$RSV_1 = (M, .16) \qquad RSV_4 = (M, 0)$$

$$RSV_6 = (VL, -.38) \qquad RSV_7 = (EL, -.36).$$

We should point out that in an ordinal context we could not distinguish the best document, however, in our linguistic IRS based on 2-tuple linguistic representation model this is possible, the best document is $d_1$.

## 4 Conclusions

In this paper, we have presented a linguistic IRS based on a 2-tuple fuzzy linguistic approach. Such a linguistic approach allows to avoid problems of loss of precision and information in the IRS results, and consequently improves its performance . This improvement is reflected in the fact that the evaluation of the documents is not only a label, but also it has associate the value of the translation from the original result to the closest index label in the linguistic term set.

Besides, we are conscious that the proposed model is complex to be used by the user. It is just a theoretical model. In future works, a graphic user interface allowing the user to utilize this model easily will be developed.

## References


[1] G. Bordogna, P. Carrara, and G. Pasi. Fuzzy Approaches to Extend Boolean Information Retrieval. In P. Bosc and J. Kacprzyk, editors, *Fuzziness in Database Management Systems*, pages 231–274. 1995.

[2] 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*, 44:70–82, 1993.

[3] G. Bordogna and G. Pasi. An Ordinal Information Retrieval Model. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(1):63–75, 2001.

[4] 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.

[5] F. Herrera and L. Martínez. A 2-tuple Fuzzy Linguistic Representation Model for Computing with Words. *IEEE Transaction on Fuzzy Systems*, 8(6):746–752, 2000.

[6] 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(1):77–88, 2001.

[7] 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):460–475, 2001.

[8] E. Herrera-Viedma, O. Cordón, M. Luque, A. G. López, and A. M. Múñoz. A Model of Fuzzy Linguistic IRS Based on Multi-Granular Linguistic Information. *International Journal of Approximate Reasoning*, 34:221–239, 2003.

[9] D.H. Kraft, G. Bordogna, and G. Pasi. An Extended Fuzzy Linguistic Approach to Generalize Boolean Information Retrieval. *Information Sciences*, 2:119–134, 1994.

[10] J.H. Lee, W.Y. Kin, M. Kim, and Y.J. Lee. On the Evaluation of Boolean Operators in the Extended Boolean Retrieval FrameWork. In *Proceedings of the 16th Annual International ACM-SIGIR*, pages 291–297, 1993.

[11] G. Miller. The Magical Number Seven, Plus mInus Two: Some Limits on Our Capacity for Processing Information. *Psychological Review*, 63:81–97, 1956.

[12] W.G. Waller and D.H. Kraft. A Mathematical Model of a Weighted Boolean Retrieval System. *Information Processing & Management*, 15:235–245, 1979.

[13] R.R Yager. A Note on Weighted Queries in Information Retrieval Systems. *Journal of the American Society for Information Science*, 38:23–24, 1987.

[14] R.R. Yager. On Ordered Weighted Averaging Aggregation Operators in Multicriteria Decision Making. *IEEE Transactions on Systems, Man, and Cybernetics*, 18:183–190, 1988.

[15] L.A. Zadeh. The Concept of a Linguistic Variable and its Applications to Approximate Reasoning. *Part I, II & III, Information Science*, 8:199–249, 8:301–157, 9:43–80, 1975.