An Information Retrieval System with Unbalanced Linguistic Information Based on the Linguistic 2-tuple Model [0]

F. Herrera, E. Herrera-Viedma
Dept. of Computer Science and A.I.
University of Granada, 18071 - Granada
herrera,viedma@decsai.ugr.es

Luis Martínez
Dept. of Computer Science.
University of Jaén, 23071 - Jaén, Spain
martin@ujaen.es

Abstract

Most information retrieval systems based on linguistic approaches use symmetrically and uniformly distributed linguistic term sets to express the weights of queries and the relevance degrees of documents. However, to improve the system-user interaction it seems more adequate to express these linguistic weights and degrees by means of unbalanced linguistic scales, i.e., linguistic term sets with different discrimination levels on both sides of mid linguistic term. In this contribution we present an information retrieval system which accepts weighted queries whose weights are expressed using unbalanced linguistic term sets. Then, system provides the retrieved documents classified in linguistic relevance classes assessed on unbalanced linguistic term sets. To do so, we use the linguistic 2-tuple model as representation base of the unbalanced linguistic information. Additionally, the linguistic 2-tuple model allows us to increase the number of relevance classes in the output of system and also to improve the performance of information retrieval system.

Keywords: information retrieval, weighted query, unbalanced linguistic term set, computing with words.

1 Introduction

Information Retrieval involves the development of computer systems for the storage and retrieval of (predominantly) textual information (documents). The main activity of an Information Retrieval System (IRS) is the gathering of the pertinent filed documents that best satisfy user information requirements (queries). Basically, IRSs present three components to carry out their activity [11]:

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

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

3.- *An Evaluation Subsystem:* which evaluates the relevance of each document for a user query by means of a retrieval status value (RSV).

A promising direction to improve the effectiveness of IRSs consists of representing in the queries the users' concept of relevance. This is a very complex task because it presents subjectivity and uncertainty. To do so, a possible solution consists in the use of weighting tools in the formulation of queries. By attaching weights in a query, a user can increase his/her expressiveness and provide a more precise description of his/her desired documents.

Different weighted IRSs based on an ordinal fuzzy linguistic approach [3, 4] were presented

[0] This work has been partially supported by Research Project PB98-1305

in [1, 2, 8, 9]. With such linguistic approach the weights are assumed qualitative values assessed on symmetrically and uniformly distributed linguistic term sets. Then, users can characterize the contents of the desired documents by explicitly associating a linguistic descriptor to a term in a query, such as "important" or "very important", and on the other hand, the estimated relevance levels of the documents are supplied in a linguistic form (e.g., linguistic terms such as "relevant", "very relevant" may be used). The problem is that using symmetrically and uniformly distributed linguistic term sets we find the same discrimination levels on both sides of mid linguistic term. However, usually users look for documents with positive criteria, that is, they formulate their weighted queries using linguistic assessments on the right of the mid label a lot more than on the left. Similarly, usually users are interested in the relevant documents a lot more than in the non-relevant documents, and then a best tuning of the output of IRS can be achieved if the IRS uses a higher number of discrimination levels on the right of the mid linguistic term. Therefore, in information retrieval the use of *unbalanced linguistic term sets* (see Figure 1) i.e., linguistic term sets with different discrimination levels on both sides of mid linguistic term, to express weighted queries and the relevance of documents seems more appropriate.

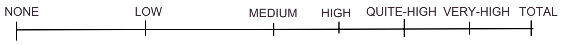

Figure 1: Unbalanced linguistic term set of 7 labels

The aim of this contribution is to present a linguistic IRS that manages unbalanced linguistic information to represent the weights of queries and the relevance degrees of retrieved documents. To do so, we use hierarchical linguistic contexts based on the linguistic 2-tuple computational model [6, 7]. In such a way, we present an IRS that improves the expressiveness in the system-user interaction. Furthermore, the use of 2-tuple model improves the performance of IRS because it increases the classification levels of relevance.

In order to do that, the contribution is structured as follows. Section 2 introduces the 2-tuple based linguistic methodology designed to manage unbalanced linguistic information. Section 3 presents the linguistic IRS. And finally, some concluding remarks are pointed out.

2 A Methodology to Manage Unbalanced Linguistic Information

In this section, we present a method to manage unbalanced linguistic information defined using the hierarchical linguistic contexts based on the linguistic 2-tuple computational model.

2.1 Linguistic Computational Model Based on 2-tuples

In [6] was presented a linguistic computational model based on linguistic 2-tuples that carries out processes of computing with words in a precise way when the linguistic term sets are symmetrically and uniformly distributed. This model is based on the concept of *symbolic translation*. It represents the linguistic information by means of linguistic 2-tuples and defines a set of functions to facilitate computational processes over 2-tuples.

Definition 1. *Let $S = \{s_0, ..., s_g\}$ be a linguistic term set and $\beta \in [0, g]$ a value supporting the result of a symbolic aggregation operation, then the 2-tuple that expresses the equivalent information to β is obtained with the following function:*

$$\Delta : [0, g] \longrightarrow S \times [-0.5, 0.5)$$
$$\Delta(\beta) = (s_i, \alpha), \quad \begin{cases} s_i & i = round(\beta) \\ \alpha = \beta - i & \alpha \in [-.5, .5) \end{cases}$$

where $round(\cdot)$ is the usual round *operation, $s_i \in S$ has the closest index label to "β" and "α" is the value of the symbolic translation.*

Proposition 1. *Let $S = \{s_0, ..., s_g\}$ be a linguistic term set and (s_i, α) be a 2-tuple. There is always a Δ^{-1} function, such that, from a 2-tuple it returns its equivalent numerical value $\beta \in [0, g] \subset \mathcal{R}$.*

Proof. It is trivial, we consider the following function:

$$\Delta^{-1} : S \times [-.5, .5) \longrightarrow [0, g]$$

$$\Delta^{-1}(s_i, \alpha) = i + \alpha = \beta$$

Remark: We should point out that the conversion of a linguistic term into a linguistic 2-tuple consists of adding a value 0 as value of symbolic translation: $s_i \in S \Longrightarrow (s_i, 0)$.

The 2-tuples linguistic computational model presents different techniques to manage the linguistic information [6]:

1.- *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 2-tuples, with each one representing a counting of information:

- if $k < l$ then (s_k, α_1) is smaller than (s_l, α_2)
- if $k = l$ then
  1. if $\alpha_1 = \alpha_2$ then (s_k, α_1), (s_l, α_2) represent the same information
  2. if $\alpha_1 < \alpha_2$ then (s_k, α_1) is smaller than (s_l, α_2)
  3. if $\alpha_1 > \alpha_2$ then (s_k, α_1) is bigger than (s_l, α_2)

2.- *Negation of 2-tuple* (s_i, α) is defined as

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

3.- *Different aggregation operators of 2-tuples*, as for example the 2-tuple arithmetic mean [6].

2.2 Hierarchical Linguistic Contexts Based on 2-tuples

The hierarchical linguistic contexts were introduced in [7] to improve the precision of processes of CW in multi-granular linguistic contexts. In this contribution, we use them to manage unbalanced linguistic term sets.

A *Linguistic Hierarchy* is a set of levels, where each level represents a linguistic term set with different granularity to the remaining levels. Each level is denoted as $l(t, n(t))$, being,

1. t a number that indicates the level of the hierarchy, and
2. *n(t)* the granularity of the linguistic term set of the level t.

We assume levels containing linguistic terms whose membership functions are triangular-shaped, symmetrically and uniformly distributed in $[0, 1]$. In addition, the linguistic term sets have an odd value of granularity.

The levels belonging to a linguistic hierarchy are ordered according to their granularity, i.e., for two consecutive levels t and $t+1$, $n(t+1) > n(t)$. Therefore, the level $t + 1$ is a refinement of the previous level t.

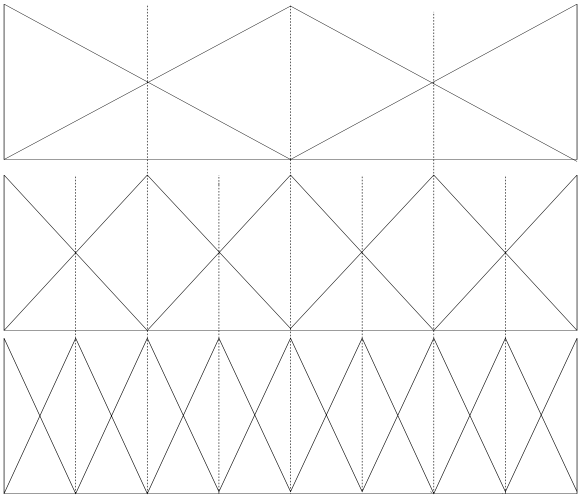

Figure 2: Linguistic Hierarchy of 3, 5 and 9 Labels

From the above concepts, we define a linguistic hierarchy, LH, as the union of all levels t:

$$LH = \bigcup_t l(t, n(t)).$$

Given an LH, we denote as $S^{n(t)}$ the linguistic term set of LH corresponding to the level t of LH characterized by a granularity of uncertainty $n(t)$:

$$S^{n(t)} = \{s_0^{n(t)}, ..., s_{n(t)-1}^{n(t)}\}.$$

Generically, we can say that the linguistic term set of level $t + 1$ is obtained from its predecessor as:

$$l(t, n(t)) \rightarrow l(t + 1, 2 \cdot n(t) - 1).$$

A graphical example of a linguistic hierarchy is shown in Figure 2.

In [7] transformation functions between labels of different levels were developed to make processes of computing with words without loss of information.

Definition 2. *Let $LH = \bigcup_t l(t,n(t))$ be a linguistic hierarchy whose linguistic term sets are denoted as $S^{n(t)} = \{s_0^{n(t)}, ..., s_{n(t)-1}^{n(t)}\}$, and let us consider the 2-tuple linguistic representation. The transformation function from a linguistic label in level t to a label in level t' is defined as:*

$$TF_{t'}^{t} : l(t,n(t)) \longrightarrow l(t',n(t'))$$

$$TF_{t'}^{t}(s_i^{n(t)}, \alpha^{n(t)}) =$$

$$\Delta_{n(t')}(\frac{\Delta_{n(t)}^{-1}(s_i^{n(t)}, \alpha^{n(t)}) \cdot (n(t')-1)}{n(t)-1}).$$

Proposition 2. The transformation function between linguistic terms in different levels of the linguistic hierarchy is bijective:

$$TF_t^{t'}(TF_{t'}^{t}(s_i^{n(t)}, \alpha^{n(t)})) = (s_i^{n(t)}, \alpha^{n(t)}).$$

2.3 A Management Method of Unbalanced Linguistic Information

Here we propose a method to manage unbalanced linguistic term sets based on the linguistic 2-tuple model. Basically, this method consists of representing unbalanced linguistic terms from different levels of an LH, carrying out computational operations of unbalanced linguistic information using the 2-tuple computational model.

The management method of unbalanced linguistic information present the following steps:

1.- *Represent the unbalanced linguistic term set $\mathcal{S}$ by means of a linguistic hierarchy, LH:*

1.1. Choose a level t^- with an adequate granularity to represent using the 2-tuple representation model the subset of linguistic terms of $\mathcal{S}$ on the left of the mid linguistic term.

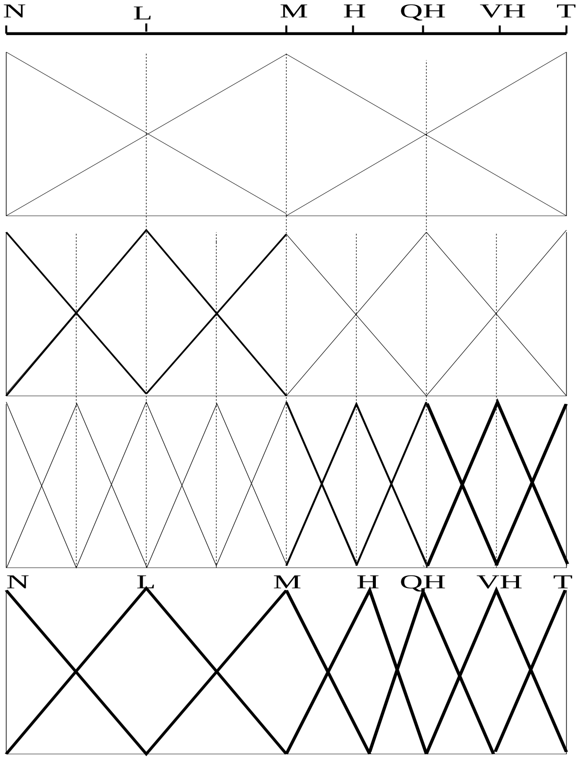

Figure 3: Representation for an Unbalanced Term Set of 7 Labels

1.2. Choose a level t^+ with an adequate granularity to represent using the 2-tuple representation model the subset of linguistic terms of $\mathcal{S}$ on the right of the mid linguistic term.

2.- *Define an unbalanced linguistic computational model:*

2.1. Choose a level $t' \in \{t^-, t^+\}$, such that $n(t') = max\{n(t^-), n(t^+)\}$.

2.2. Define the comparison of two 2-tuples $(s_k^{n(t)}, \alpha_1)$, $t \in \{t^-, t^+\}$, and $(s_l^{n(t)}, \alpha_2)$, $t \in \{t^-, t^+\}$, with each one representing a counting of unbalanced information. Its expression is similar to the usual comparison of two 2-tuples but acting on the values $TF_{t'}^{t}(s_k^{n(t)}, \alpha_1)$ and $TF_{t'}^{t}(s_l^{n(t)}, \alpha_2)$. We should point out that using the comparison of 2-tuples we can easily define the comparison operators Max and Min.

2.3 Define the negation operator of unbalanced linguistic information. Let $(s_k^{n(t)}, \alpha)$, $t \in \{t^-, t^+\}$ be an unbalanced 2-tuple then:

$$\mathcal{NEG}(s_k^{n(t)}, \alpha) = Neg(TF_{t''}^{t}(s_k^{n(t)}, \alpha)),$$

$t \neq t'', t'' \in \{t^-, t^+\}$.

2.4. Define aggregation operators of unbalanced linguistic information. This is done using the aggregation processes designed in the 2-tuple computational model but acting on the unbalanced linguistic values transformed by means of $TF^t_{t'}$. Then, once it is obtained a result, it is transformed to the correspondent level t by means of $TF^{t'}_t$ for expressing the result in the unbalanced linguistic term set.

Assuming the unbalanced linguistic term set shown in Figure 1 and the linguistic hierarchy shown in Figure 2, in Figure 3 we show how to select the different levels to represent the unbalanced linguistic term set.

3 The IRS with Unbalanced Linguistic Information

In this section we present a linguistic IRS which uses an unbalanced linguistic term set $\mathcal{S}$ to express the linguistic assessments in the retrieval process. Particularly, $\mathcal{S}$ presents a higher number of discrimination levels on the right of the mid linguistic term than on the left (e.g. as happens in Figure 1). Then, this IRS accepts linguistic weighted queries and provides linguistic retrieval status values (RSVs) assessed on $\mathcal{S}$ and $\mathcal{S} \times [-.5, .5]$, respectively. The components of this IRS are presented in the following subsections.

3.1 Database

The database stores the finite set of documents D= $\{d_1, \ldots, d_m\}$ and the finite set of index terms T= $\{t_1, \ldots, t_l\}$. Documents are represented by means of index terms, which describe the subject content of the documents. A numeric indexing function F : $\mathrm{D} \times \mathrm{T} \rightarrow [0, 1]$, exists. F weighs index terms according to their significance in describing the content of a document in order to improve the retrieval of documents. $F(d_j, t_i) = 0$ implies that the document d_j is not at all about the concept(s) represented by index term t_i and $F(d_j, t_i) = 1$ implies that the document d_j is perfectly represented by the concept(s) indicated by t_i. Then each d_j is represented as $R_{d_j} = \Sigma^l_{i=1} F(d_j, t_i)/t_i$.

We assume that the system uses any of the existing weighting methods [11] to compute F.

3.2 The Query Subsystem

The query subsystem accepts Boolean queries whose terms can be weighted simultaneously by means of linguistic threshold weights and linguistic importance weights taken from a linguistic term set $\mathcal{S}$, assuming as reference domain $U = [0, 1]$. By associating threshold weights with terms in a query, the user is asking to see all the documents sufficiently about the topics represented by such terms. By associating importance weights to terms in a query, the user is asking to see all documents whose content represents the concept that is more associated with the most important terms than with the less important ones. Then, each query is expressed as a combination of the weighted index terms which are connected by the logical operators AND ($\wedge$), OR ($\vee$), and NOT ($\neg$). Therefore, a query is any legitimate Boolean expression whose atomic components (atoms) are 3-tuples $< t_i, c^1_i, c^2_i >$ belonging to the set, T $\times \mathcal{S}^2$; $t_i \in$ T, and c^1_i and c^2_i are linguistic values of the linguistic variable *Importance* modeling the threshold (importance that the term t_i must have in the desired documents) and importance semantic (importance that the meaning of t_i must have in the set of retrieved documents), respectively. We must point out that the importance semantics plays a role in the aggregation phase of the evaluation of a compound query. By simplifying, we assume that user queries are preprocessed and put into a disjunctive normal form (DNF).

3.3 The Evaluation Subsystem

The evaluation subsystem evaluates weighted queries by means of a constructive bottom-up process which includes two steps. Firstly, the documents are evaluated according to their relevance only to atoms of the query. In this first step, the threshold semantic is applied. Secondly, the documents are evaluated according to their relevance to Boolean combinations of atomic components, and so on, working in a bottom-up fashion until the whole query is processed. In this second step, the importance semantic is applied. At the end a total linguistic RSV, which is a 2-tuple taken from $\mathcal{S} \times [-0.5, 0.5]$ is assigned to each document with respect to the whole query. Then, the evaluation subsystem presents the retrieved documents arranged in linguistic relevance classes as in [1, 8, 9], but in this case, it is developed a better tuning for IRS response because the use of 2-tuple representation model increases the number of classification levels. We synthesize the evaluation subsystem using a linguistic evaluation function E as in [8], but defined linguistically on $\mathcal{S} \times [-0.5, 0.5]$, i.e.,

$$E\text{:Q} \times \text{D} \to \mathcal{S} \times [-0.5, 0.5],$$

being Q the set of all legitimate queries. E acts according to the following four rules for all $d_j \in D$:

1.- Evaluation of a simple query with one atom $q_i =< t_i, c_i^1, c_i^2 >$:

$$E(q_i, d_j) = g(\Delta(n(t) \cdot F(d_j, t_i)), (c_i^1, 0))$$

with

$$\Delta : [0, n(t) - 1] \to \mathcal{S} \times [-0.5, 0.5]$$

$t = t^-$ if $F(d_j, t_i) \leq 0.5$ and $t = t^+$ if $F(d_j, t_i) > 0.5$; and being g the matching function for threshold semantic proposed in [10], but defined linguistically as $g : [\mathcal{S} \times [-0.5, 0.5]]^2 \to \mathcal{S} \times [-0.5, 0.5]$

$$g((s_v, r_v), (s_w, r_w)) = \begin{cases} (s_v, r_v) & \text{if } (s_v, r_v) \geq (s_w, r_w) \\ \Delta(0) & \textit{otherwise.} \end{cases}$$

We consider that the importance semantic has not sense in a simple query as in [8].

2.- Evaluation of an And query $p_k = q_1 \wedge \ldots \wedge q_n (n \geq 2)$:

$$E(p_1, d_j) = Min(Max(E(q_1, d_j), \mathcal{NEG}(c_1^2, 0)), \ldots, Max(E(q_n, d_j), \mathcal{NEG}(c_n^2, 0))).$$

3.- Evaluation of an query in DNF $p = \vee_{k=1}^{(m \geq 2)} p_k$:

$$E(p, d_j) = Max((E(p_1, d_j), \ldots, (E(p_m, d_j)).$$

4.- Evaluation of an negated query $\neg p$:

$$E(\neg p, d_j) = \mathcal{NEG}(E(p, d_j)).$$

4 Concluding Remarks

In this contribution we have presented a linguistic IRS using unbalanced linguistic term sets. In such a way, on the one hand, users can use a higher number of discrimination values to assess the importance assigned to the terms of queries, and on the other hand, the system has also a higher number of discrimination values to assess the relevance assigned to the retrieved documents. To do so, we have developed a methodology to manage unbalanced linguistic information based on the linguistic 2-tuple representation model and the linguistic hierarchical contexts. Additionally, this methodology allows us to improve the performance of IRS by increasing the classification levels of the documents.

References

[1] G. Bordogna and G. Pasi, Application of the OWA Operators to Soften Information Retrieval Systems, in: R.R Yager and J. Kacprzyk, Eds., *The Ordered Weighted Averaging Operators: Theory and Applications* (Kluwer Academic Publishers, 1997) 275-294.

[2] G. Bordogna and G. Pasi, An Ordinal Information Retrieval Model. *Int. J. of Uncertainty, Fuzziness and Knowledge-Based Systems,* **9** (2001) 63-76.

[3] M. Delgado, J.L. Verdegay and M.A. Vila, On Aggregation Operations of Linguistic Labels. *Int. J. of Intelligent Systems,* **8** (1993) 351-370.

[4] F. Herrera and E. Herrera-Viedma, Linguistic Decision Analisys: Steps for Solving Decision Problems under Linguistic Information. *Fuzzy Sets and Systems,* **115** (2000) 67-82.

[5] 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** (1996) 175-190.

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

[7] F. Herrera and L. Martínez, A Model Based on Linguistic 2-tuples for Dealing with Multigranularity Hierarchical Linguistic Contexts in Multiexpert Decision-Making. *IEEE Trans. on Systems, Man and Cybernetics. Part B: Cybernetics,* **31:2** (2001) 227-234.

[8] E. Herrera-Viedma, Modeling the Retrieval Process for an Information Retrieval System Using an Ordinal Fuzzy Linguistic Approach. *Journal of the American Society for Information Science and Technology,* **52:6** (2001) 460-475.

[9] E. Herrera-Viedma, An IR model with Ordinal Linguistic Weighted Queries Based on Two Weighting Elements. *Int. J. of Uncertainty, Fuzziness and Knowledge-Based Systems,* **9** (2001) 77-88.

[10] T. Radecki, Fuzzy Set Theorical Approach to Document Retrieval. *Information Processing & Management,* **15** (1979) 247-260.

[11] G. Salton and M.H. McGill, *Introduction to Modern Information Retrieval.* (New York: McGraw-Hill, 1983).