Available online at www.sciencedirect.com
SCIENCE DIRECT
ELSEVIER
Fuzzy Sets and Systems 148 (2004) 61–83
FUZZY sets and systems
www.elsevier.com/locate/fss

# Incorporating filtering techniques in a fuzzy linguistic multi-agent model for information gathering on the web

E. Herrera-Viedma[a,*], F. Herrera[a], L. Martínez[b], J.C. Herrera[a], A.G. López[a]

[a] *Department of Computer Science and Artificial Intelligence, Library Science Studies School, University of Granada, Granada 18071, Spain*

[b] *Department of Computer Science, University of Jaén, Jaén 23071, Spain*

**Abstract**

In (Computing with Words, Wiley, New York, 2001, p. 251; Soft Comput. 6 (2002) 320; Fuzzy Logic and The Internet, Physica-Verlag, Springer, Wurzburg, Berlin, 2003) we presented different fuzzy linguistic multi-agent models for helping users in their information gathering processes on the Web. In this paper we describe a new fuzzy linguistic multi-agent model that incorporates two information filtering techniques in its structure: a content-based filtering agent and a collaborative filtering agent. Both elements are introduced to increase the information filtering possibilities of multi-agent system on the Web and, in such a way, to improve its retrieval issues.


*Keywords:* Web; Information retrieval; Intelligent agents; Filtering; Fuzzy linguistic modelling

## 1. Introduction

The networked world contains a vast amount of data. The exponential increase in Web sites and Web documents is contributing to that Internet users not being able to find the information they seek in a simple and timely manner. Users are in need of tools to help them cope with the large amount of information available on the Web [21,22]. Therefore, techniques for searching and mining the Web are becoming increasing vital. Two important techniques that have been addressed in improving the information access on the Web are related to *intelligent agents* and *information filtering*.

Intelligent agents applied on the Web deal with the information gathering process assisting Internet users to find the fittest information to their needs [3,9,20,32]. Usually, several intelligent agents (e.g. interface agent, information discovery agent) organized in distributed architectures take part in the

* Corresponding author. Tel.: +34-95824-4258; fax: +34-95824-3317.
*E-mail address:* viedma@decsai.ugr.es (E. Herrera-Viedma).

0165-0114/$ - see front matter 
doi:10.1016/j.fss.2004.03.006

information gathering activity [3,8,9,20,25]. The problem is the design of appropriate communication protocols among the agents. The great variety of representations and evaluations of the information in the Internet is the main obstacle to this communication, and the problem becomes more noticeable when users take part in the process. This reveals the need of more flexibility in the communication among agents and between agents and users [7,31–33]. To solve this problem we presented in [6,7,12] different distributed intelligent agent models based on *fuzzy linguistic information*. Using different fuzzy linguistic approaches [13,14,34], in [6,7,12] we proposed to introduce and handle flexible information by means of linguistic labels, and in such a way, improving communication processes.

Another promising direction to improve the information access on the Web concerns the way in which it is possible to filter the great amount of information available across the Web. Information filtering is a name used to describe a variety of processes involving the delivery of information to people who need it. Operating in textual domains, *filtering systems* or *recommender systems* evaluate and filter the great amount of information available on the Web (usually, stored in HTML or XML documents) to assist people in their search processes [27]. Traditionally, these systems have fallen into two main categories [26]. *Content-based filtering systems* filter and recommend the information by matching user query terms with the index terms used in the representation of documents, ignoring data from other users. These recommender systems tend to fail when little is known about user information needs, e.g. as happens when the query language is poor. *Collaborative filtering systems* use explicit or implicit preferences from many users to filter and recommend documents to a given user, ignoring the representation of documents. These recommender systems tend to fail when little is known about a user, or when he/she has uncommon interests [26]. Several researchers are exploring hybrid content-based and collaborative recommender systems to smooth out the disadvantages of each one of them [1,4,10,26]. Applications of hybrid-based recommender systems on the Web include search tools such as Google (www.google.com) and Inquirus 2 (inquirus.nj.nec.com/i2/inq2.pl) that combines results of both content searches and collaborative recommendations. Recommender systems employing information filtering techniques often do so through the use of information filtering agents [29]. Operating in the domain of Usenet news, NewT [24] employs a vector-space based genetic algorithm to learn which articles should be selected an which should not. RE:Agent [2] use learning techniques to classify e-mail based on a user's prior actions. Finally, Amalthaea [25] is a multi-agent system for recommending Web sites.

In this paper, we present a new fuzzy linguistic multi-agent model for information gathering on the Web that uses different information filtering techniques to improve retrieval issues. We design it by using a 2-tuple fuzzy linguistic approach [14,15] as a way to endow the retrieval process with a higher flexibility, uniformity and precision. As we did in [6], the communication of the evaluation of the retrieved information among the agents is carried out by using linguistic information represented by the 2-tuple fuzzy linguistic representation. The main novelty of this multi-agent model is that it combines both content-based filtering and collaborative filtering techniques. Users represent their information needs by means of linguistic multi-weighted queries [17] and providing an information need category (medicine, decision making, economy). The multi-weighted queries are composed of terms which are weighted simultaneously by means of both linguistic threshold weights and linguistic relative importance weights. To exploit user preferences the multi-agent model incorporates two new elements in its architecture: (i) a *content-based information filtering agent* that filters the documents by mean of a matching function used to model the threshold weights, and (ii) a *collaborative filtering* *agent* that filters and recommends documents related to information need category according to the evaluation judgements previously expressed by other users.

The paper is structured as follows. Section 2 reviews the 2-tuple fuzzy linguistic representation. Section 3 presents the new fuzzy linguistic multi-agent model based on information filtering techniques. Section 4 presents an example for illustrating our proposal. Finally, some concluding remarks are pointed out.

## 2. The 2-tuple fuzzy linguistic approach

The use of Fuzzy Sets Theory has given very good results for modelling qualitative information [34]. It is a technique that handles fuzziness and represents qualitative aspects as linguistic labels by means of "linguistic variables", that is, variables whose values are not numbers but words or sentences in a natural or artificial language.

The 2-tuple fuzzy linguistic approach was introduced in [14–16] to overcome the problems of loss of information of other fuzzy linguistic approaches [11,13,34]. Its main advantage is that the linguistic computational model based on linguistic 2-tuples can carry out processes of computing with words easier and without loss of information.

### *2.1. The concept of symbolic translation and 2-tuple representation model*

Let $S=\{s_0,\dots,s_g\}$ be a linguistic term set with odd cardinality ($g+1$ is the cardinality of $S$ and usually is equal to 7 or 9), where the mid term represents an assessment of approximately 0.5 and with the rest of the terms being placed symmetrically around it. We assume that the semantics of labels is given by means of triangular membership functions represented by a 3-tuple $(a,b,c)$ and consider all terms distributed on a scale on which a total order is defined $s_i \leqslant s_j \Leftrightarrow i \leqslant j$. An example may be the following set of seven terms (Fig. 1):

$$
\begin{aligned}
&s_0 = Null(N) = (0,0,0.17), \quad s_1 = VeryLow(VL) == (0,0.17,0.33),\\
&s_2 = Low(L) = (0.17,0.33,0.5), \quad s_3 = Medium(M) = (0.33,0.5,0.67),\\
&s_4 = High(H) = (0.5,0.67,0.83), \quad s_5 = VeryHigh(VH) = (0.67,0.83,1),\\
&s_6 = Perfect(P) = (0.83,1,1).
\end{aligned}
$$

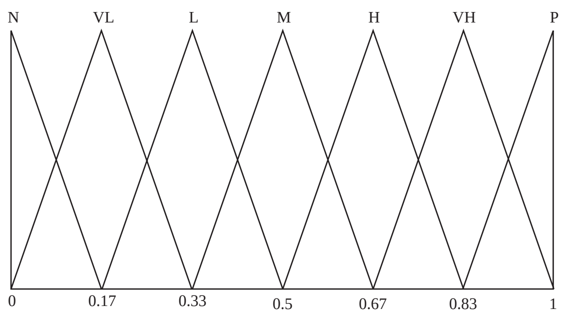


Fig. 1. A set of seven linguistic terms with its semantics.

In this fuzzy linguistic context, if a symbolic method [11,13] aggregating linguistic information obtains a value $\beta \in [0, g]$, and $\beta \notin \{0, \ldots, g\}$, then an approximation function is used to express the result in $S$.

**Definition 1** (Herrera and Martinez [14]). Let $\beta$ be the result of an aggregation of the indexes of a set of labels assessed in a linguistic term set $S$, i.e., the result of a symbolic aggregation operation, $\beta \in [0, g]$. Let $i = round(\beta)$ and $\alpha = \beta - i$ be two values, such that, $i \in [0, g]$ and $\alpha \in [-0.5, 0.5)$ then $\alpha$ is called a *Symbolic Translation*.

Roughly speaking, the symbolic translation of a linguistic term, $s_i$, is a numerical value assessed in $[-0.5, 0.5)$ that supports the "difference of information" between a counting of information $\beta \in [0, g]$ obtained after a symbolic aggregation operation and the closest value in $\{0, \ldots, g\}$ that indicates the index of the closest linguistic term in $S$ ($i = round(\beta)$).

The 2-tuple fuzzy linguistic approach is developed from the concept of symbolic translation by representing the linguistic information by means of 2-tuples $(s_i, \alpha_i)$, $s_i \in S$ and $\alpha_i \in [-0.5, 0.5)$:

- $s_i$ represents the linguistic label of the information, and
- $\alpha_i$ is a numerical value expressing the value of the translation from the original result $\beta$ to the closest index label, $i$, in the linguistic term set ($s_i \in S$).

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

**Definition 2** (Herrera and Martinez [14]). Let $S = \{s_0, \ldots, s_g\}$ be a linguistic term set and $\beta \in [0, g]$ a value representing the result of a symbolic aggregation operation, then the 2-tuple that expresses the equivalent information to $\beta$ is obtained with the following function:

$$\Delta : [0, g] \to S \times [-0.5, 0.5)$$

$$\Delta(\beta) = (s_i, \alpha) \quad \text{with} \quad \begin{cases} s_i & i = round(\beta) \\ \alpha = \beta - i & \alpha \in [-0.5, 0.5) \end{cases}$$

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

In [14] we show that there exists $\Delta^{-1}$, such that, from a 2-tuple $(s_i, \alpha)$ it returns its equivalent numerical value $\beta \in [0, g] \subset \mathscr{R}$, which is obtained as $\Delta^{-1}(s_i, \alpha) = i + \alpha$. On the other hand, it is obvious that the conversion of a linguistic term into a linguistic 2-tuple consists of adding a symbolic translation value of $0 : s_i \in S \Rightarrow (s_i, 0)$.

### 2.2. 2-tuple linguistic computational model

The 2-tuple linguistic computational model is defined by presenting the comparison of 2-tuples, a negation operator and aggregation operators of 2-tuples.

1. *Comparison of 2-tuples*: The comparison of linguistic information represented by 2-tuples is carried out according to an ordinary lexicographic order. Let $(s_k, \alpha_1)$ 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 $(s_l,\alpha_2)$.

2. *Negation operator of 2-tuples*: This operator is defined as follows:

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

3. *Aggregation operators of 2-tuples*: The aggregation of information consists of obtaining a value that summarizes a set of values, therefore, the result of the aggregation of a set of 2-tuples must be a 2-tuple. In the literature we can find many aggregation operators which allow us to combine the information according to different criteria. Using functions $\Delta$ and $\Delta^{-1}$ that transform without loss of information numerical values into linguistic 2-tuples and viceversa, any of the existing aggregation operator can be easily extended for dealing with linguistic 2-tuples. Some examples are

- *Arithmetic Mean*. The arithmetic mean is a classical numerical aggregation operator. Its equivalent operator, for linguistic 2-tuples, is defined as,

**Definition 3.** Let $x=\{(r_1,\alpha_1),\dots,(r_n,\alpha_n)\}$ be a set of linguistic 2-tuples, the 2-tuple arithmetic mean $\bar{x}^{\mathrm{e}}$ is computed as,

$$\bar{x}^{\mathrm{e}}[(r_1,\alpha_1),\dots,(r_n,\alpha_n)] = \Delta\left(\sum_{i=1}^{n}\frac{1}{n}\Delta^{-1}(r_i,\alpha_i)\right) = \Delta\left(\frac{1}{n}\sum_{i=1}^{n}\beta_i\right).$$

- *Weighted average operator.* The weighted average is used when different values $x_i$ have a different importance in the nature of the variable $x$. To do so, each value $x_i$ has a weight associated to it, $w_i$, indicating its importance in the nature of the variable. The equivalent operator for linguistic 2-tuples is defined as:

**Definition 4.** Let $x=\{(r_1,\alpha_1),\dots,(r_n,\alpha_n)\}$ be a set of linguistic 2-tuples and $W=\{w_1,\dots,w_n\}$ be their associated weights. The 2-tuple weighted average $\bar{x}^w$ is

$$\bar{x}^w[(r_1,\alpha_1),\dots,(r_n,\alpha_n)] = \Delta\left(\frac{\sum_{i=1}^{n}\Delta^{-1}(r_i,\alpha_i)\cdot w_i}{\sum_{i=1}^{n}w_i}\right) = \Delta\left(\frac{\sum_{i=1}^{n}\beta_i\cdot w_i}{\sum_{i=1}^{n}w_i}\right).$$

- *Linguistic weighted average operator*. This operator is an extension of $\bar{x}^w$ assuming that the weights are expressed by means of linguistic 2-tuples.

**Definition 5.** Let $x=\{(r_1,\alpha_1),\ldots,(r_n,\alpha_n)\}$ be a set of linguistic 2-tuples and $W=\{(w_1,\alpha_1^w),\ldots,(w_n,\alpha_n^w)\}$ be their linguistic 2-tuple associated weights. The 2-tuple linguistic weighted average $\bar{x}_l^w$ is

$$\bar{x}_l^w[((r_1,\alpha_1),(w_1,\alpha_1^w))\ldots((r_n,\alpha_n),(w_n,\alpha_n^w))] = \Delta\left(\frac{\sum_{i=1}^n \beta_i \cdot \beta_{W_i}}{\sum_{i=1}^n \beta_{W_i}}\right),$$

with $\beta_i = \Delta^{-1}(r_i,\alpha_i)$ and $\beta_{W_i} = \Delta^{-1}(w_i,\alpha_i^w)$.

## 3. A fuzzy linguistic multi-agent model based on information filtering techniques

In this section we present a new fuzzy linguistic multi-agent model based on a 2-tuple fuzzy linguistic approach. It is developed from the multi-agent model defined in [6]. We propose to improve the performance of that by incorporating in its architecture both content-based filtering and collaborative filtering techniques.

### *3.1. Architecture of the multi-agent model presented in [6]*

A multi-agent system is one in which a number of agents cooperates and interact with each other in a complex and distributed environment. In a typical multi-agent system the agents work together to achieve a global objective based on distributed data and control. Multi-agent systems have been widely used in Web applications [5,23,25,29]. In [9,20] a detailed study on multi-agent system is presented.

In [30] a distributed multi-agent model for the information gathering is defined. This model develops the retrieval activity by considering five action levels: *internet users*, *interface agents*, *task agents*, *information agents and information sources*. Using this model, in [6] we defined a fuzzy linguistic distributed multi-agent model that uses linguistic 2-tuples to carry out the communication processes among the agents. In such a way, we incorporate in the retrieval process a higher degree of flexibility to carry out the information interchange, but in a precise way. This model presents a hierarchical architecture with five activity levels:

- *Level 1*: *Internet user*, which looks for Web documents on the Internet by means of a weighted query where a set of terms $\{t_1,t_2,\ldots,t_m\}$ related to the desired documents is specified together with their respective linguistic relative importance degrees $\{p_1,p_2,\ldots,p_m\}$, $p_i \in S$.
- *Level 2*: *Interface agent* (generally one for user), that communicates the user weighted query to the task agent, and filters the retrieved documents from task agent in order to give the user those ones that better satisfy his/her needs.
- *Level 3*: *Task agent* (generally one for interface agent), that communicates the terms of user query to the information agents, and get those documents from every information agent that better fulfill the weighted query, fusing them and resolving the possible conflicts among the information agents.
- *Level 4*: *Information agents*, which receive the terms of user query from the task agent and look for the documents in the information sources. Then, the task agent receives from every information agent $h$ a set of documents and their relevance $(D^h,R^h)$, where every document $d_j^h$ has an associated degree of relevance $r_j^h \in [0,1]$ $(j=1,\ldots,\#(D^h))$. It also receives a set of

linguistic degrees of satisfaction $C^h = \{c_1^h, c_2^h, \ldots, c_m^h\}$, $c_i^h \in S \times [-0.5, 0.5)$ of this set of documents with regard to every term of the query.

- *Level 5*: *Information sources*, consisting of all data sources within the Internet, such as databases and information repositories.

The architecture of this model in the case of a single user scheme is represented in Fig. 2.

### 3.2. Architecture of fuzzy linguistic multi-agent model based on information filtering techniques

As it is known, a promising direction to improve the effectiveness of search engines concerns the way in which it is possible to "filter" the great amount of information available across the Internet [19]. As it was said at the beginning, the so-called recommender systems are useful tools to carry out the evaluation and filtering activities on the Web [27]. The combined use of recommender systems together with search multi-agent systems has given very good issues on the Web [2,24,25,29].

Then, our idea consists of applying the use of recommender systems in the multi-agent model presented in [6] to improve its performance. The incorporation of recommender systems in its architecture increases its information filtering possibilities on the Web. To do so, we present a new fuzzy linguistic multi-agent model that combines in its activity the two more important existing filtering techniques, content-based filtering and collaborative filtering [26,27]. It integrates in its architecture two new levels: the level of the *content-based filtering agents* and the level of *collaborative filtering agent*. Furthermore, the users' expression possibilities are increased. Users specify their information needs by means of both a linguistic multi-weighted query and an information need category. Multi-weighted query languages allow user to express better their ideas of concept of relevance and, in such a way, information retrieval systems have more possibilities to find their desired documents [17,18]. Each term of a user query can be weighted simultaneously by two linguistic weights. The first weight is associated with a classical threshold semantics and the second one with a relative importance semantics. By associating threshold weights with terms in a query, the user is asking to see all the documents sufficiently related to the topics represented by such terms. The threshold weights are used by the content-based filtering agents to carry out a first filtering of documents to retrieve. By associating relative 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 rather than with the least important ones. The relative importance weights are used by the task agent to determinate the number of documents to be retrieved from each content-based filtering agent. The information need category represents the interest topic of the user's information need, e.g.,"information retrieval", "medicine", "decision making". Previously, a list of information categories available to users must be established. The information need category is used by the collaborative filtering agent to carry out a second filtering of documents that are retrieved and shown to the users definitively.

This new multi-agent model presents a hierarchical architecture that contains 7 activity levels (see Fig. 3):

- *Level 1*: *Internet user*, which expresses his/her information needs by means of a linguistic multi-weighted query $\{(t_1, p_1^1, p_1^2), (t_2, p_2^1, p_2^2), \ldots, (t_m, p_m^1, p_m^2)\}$, $p_i^1, p_i^2 \in S$ and an information need category $\mathscr{A}_i \in \{\mathscr{A}_1, \ldots, \mathscr{A}_l\}$. He also provides his/her identity $\mathscr{I}\mathscr{D}$ (e.g. e-mail).

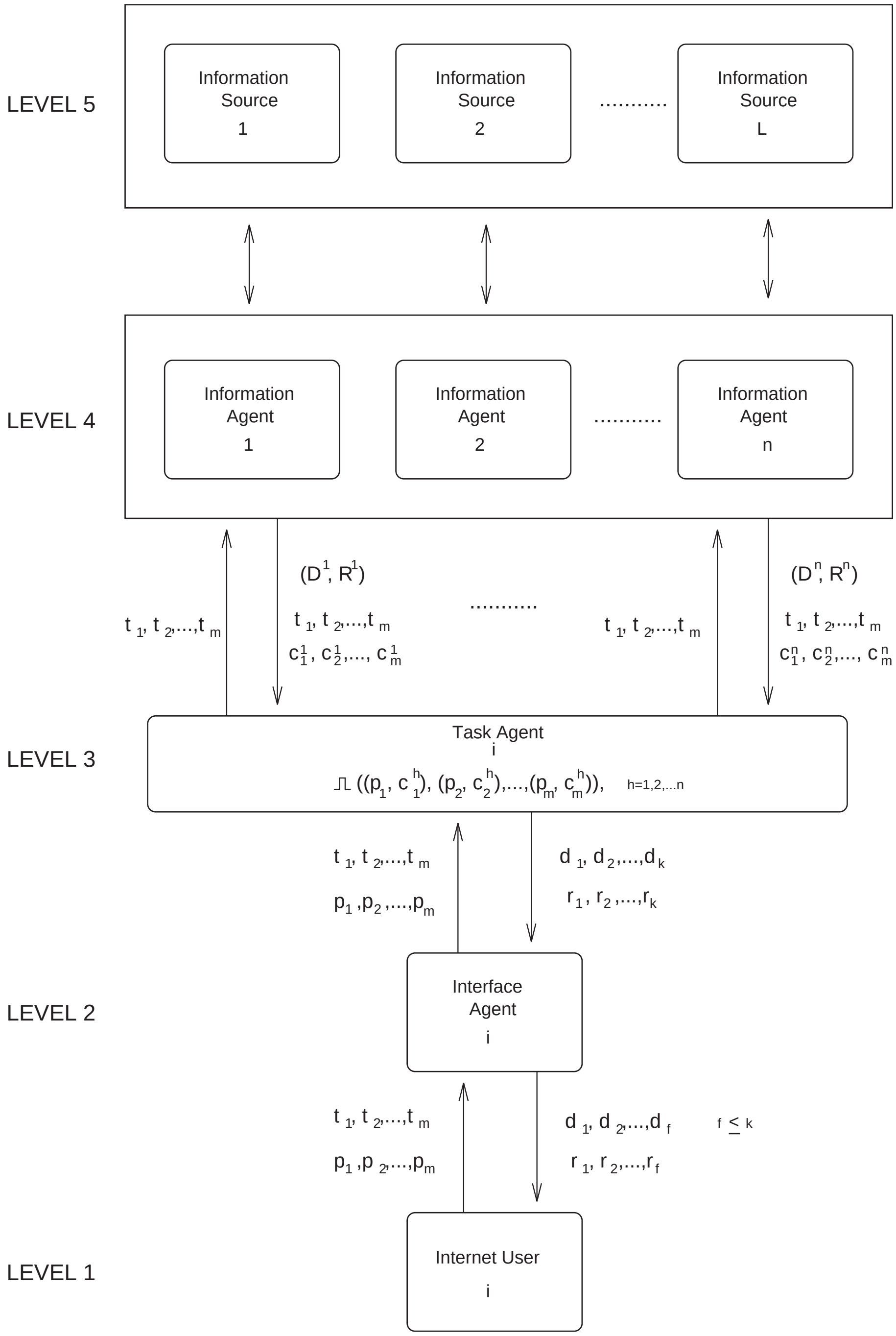


Fig. 2. Structure of multi-agent model [6].

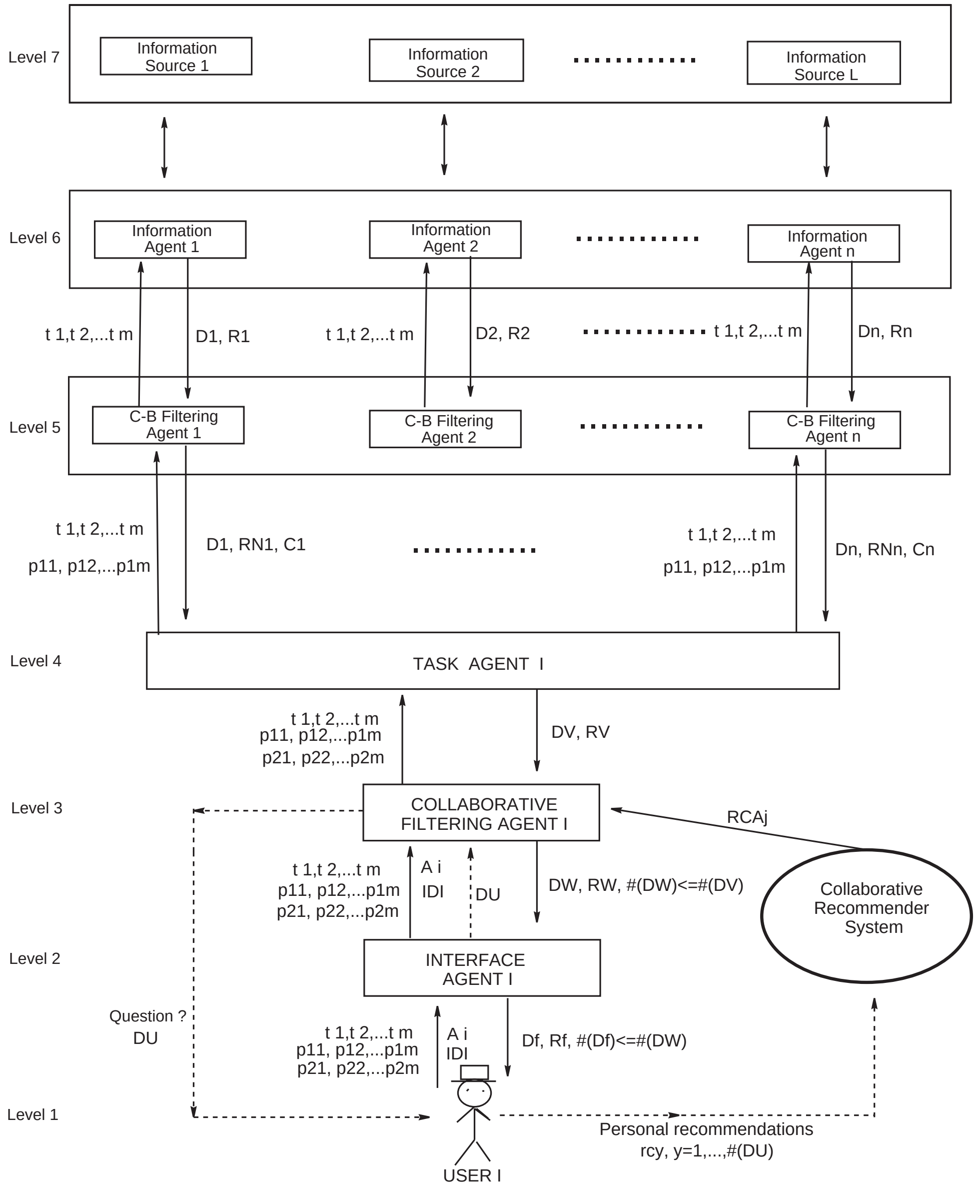


Fig. 3. Structure of a multi-agent model based on filtering agents.

- *Level 2*: *Interface agent* (one for user), that communicates the user multi-weighted query, the information need category and the user identity to the collaborative filtering agent, filters the retrieved documents from collaborative filtering agent to give to the user those that satisfy better his/her needs, and finally, informs the collaborative filtering agent on set of documents used by user to satisfy his/her information needs $DU$.

- *Level 3*: *Collaborative filtering agent* (one for interface agent), that communicates the user multi-weighted query to the task agent, receives the more relevant documents chosen by the task agent, retrieves the recommendations on such documents from a collaborative recommendation system using the information need category expressed by the user $RC^{\mathscr{A}_i} = \{RC_1^{\mathscr{A}_i}, \ldots, RC_v^{\mathscr{A}_i}\} RC_j^{\mathscr{A}_i} \in S \times [-0.5, 0.5)$, filters the documents by recalculating their relevance using these recommendations, and communicates these documents together with their new relevance degrees to the interface agent. Later, it carries out the tasks to update in the collaborative recommendation system the recommendations on the documents used by the user, i.e., it invites user to provide a recommendation $rc_y$ on each chosen document $d_y^U \in DU$ and this recommendation is stored in the collaborative recommendation system together with the recommendations provided by other users that used $d_y^U$.
- *Level 4*: *Task agent* (one for collaborative filtering agent), that communicates the terms of user query together with their respective threshold weights to the content-based filtering agents, and filters documents provided by content-based filtering agents by getting those documents from every content-based filtering agent that fulfill better the weighted query, fusing them and resolving the possible conflicts among the content-based filtering agents.
- *Level 5*: *Content-based filtering agents* (one for information agent). Each content-based filtering agent communicates the terms of user query to its respective information agent and filters the relevant documents provided by its information agent by recalculating their relevance using the threshold weights. Then, the task agent receives from every content-based filtering agent $h$ a set of documents and their relevance $(D^h, RN^h)$, where every document $d_j^h$ has associated a linguistic degree of relevance expressed in linguistic 2-tuples $rn_j^h \in S \times [-0.5, 0.5)$ $(j = 1, \ldots, \#(D^h))$. It also receives a set of linguistic degrees of satisfaction $C^h = \{c_1^h, c_2^h, \ldots, c_m^h\}$, $c_i^h \in S \times [-0.5, 0.5)$ of this set of documents $D^h$ with regard to every term of the query $t_i$.
- *Level 6*: *Information agents*, which receive the terms of user query from the content-based filtering agents and look for the documents in the information sources. Then, each content-based filtering agent $h$ receives from its respective information agent $h$ the set of relevant documents that it found through information sources $D^h$ and their relevance $R^h$, where every document $d_j^h$ has an associated degree of relevance $r_j^h \in S \times [-0.5, 0.5)$ $(j = 1, \ldots, \#(D^h))$.
- *Level 7*: *Information sources.*

### 3.3. Operation of fuzzy linguistic multi-agent model based on filtering agents

The activity of multi-agent model presented in the above subsection is composed of two phases:

1. *Retrieval phase*: This first phase coincides with the information gathering process developed by the multi-agent model itself, i.e., this phase begins when a user specifies his/her query and finishes when he/she chooses his/her desired documents among the relevant documents retrieved and provided by the system.

2. *Feedback phase*: This second phase coincides with the updating process of collaborative recommendations on desired documents existing in the collaborative recommender system, i.e., this phase begins when the *interface agent* informs the documents chosen by the user to the *collaborative filtering agent* and finishes when the recommender system recalculates and updates the recommendations of the desired documents.

#### 3.3.1. Retrieval phase

The information gathering process of multi-agent model is carried out as follows:

- *Step 1*: An *Internet user* expresses his/her information needs by means of a linguistic multi-weighted query $\{(t_1, p_1^1, p_1^2), (t_2, p_2^1, p_2^2), \ldots, (t_m, p_m^1, p_m^2)\}$, $p_i^1, p_i^2 \in S$ and an information need category $\mathscr{A}_i$ chosen from a list of information need categories $\{\mathscr{A}_1, \ldots, \mathscr{A}_l\}$ provided by the system. The system also requires the user's identity $\mathscr{I}\mathscr{D}$. All this information is given by the user to the *interface agent*.
- *Step 2*: The *interface agent* gives the linguistic multi-weighted query together with the information need category to the *collaborative filtering agent*.
- *Step 3*: The *collaborative filtering agent* gives the linguistic multi-weighted query to the *task agent*.
- *Step 4*: The *task agent* communicates the terms of the query $\{t_1, t_2, \ldots, t_m\}$ together with their respective linguistic threshold weights $\{p_1^1, p_2^1, \ldots, p_m^1\}$, $p_i^1 \in S$ to all the *content-based filtering agents* to which it is connected.
- *Step 5*: Each *content-based filtering agent h* makes the query to its respective *information agent h* and gives it the terms of the query $\{t_1, t_2, \ldots, t_m\}$.
- *Step 6*: All the *information agents* that have received the query, look for the information that better satisfies it in the *information sources*, and retrieve from them the documents. We assume that the documents are represented in the *information sources* using an index term based representation as in Information Retrieval [17,18,28]. Then, there exists a finite set of index terms $T = \{t_1, \ldots, t_l\}$ used to represent the documents and each document $d_j$ is represented as a fuzzy subset

$$d_j = \{(t_1, F(d_j, t_1)), \ldots, (t_l, F(d_j, t_l))\}, \quad F(d_j, t_i) \in [0, 1],$$

where $F$ is any numerical indexing function that weighs index terms according to their significance in describing the content of a document. $F(d_j, t_i) = 0$ 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 $t_i$.
- *Step 7*: Each *content-based filtering agent h* receives from its respective *information agent h* a set of documents and their relevance $(D^h, R^h)$ ordered decreasingly by relevance. Every document $d_j^h$ has associated a linguistic degree of relevance $r_j^h \in S \times [-0.5, 0.5)$, which is calculated as

$$r_j^h = \bar{x}^{\mathrm{e}}[\Delta(g \cdot F(d_j^h, t_1)), \ldots, \Delta(g \cdot F(d_j^h, t_m))] = \Delta\left(g \cdot \sum_{i=1}^{m} \frac{1}{m} F(d_j^h, t_i)\right),$$

$g + 1$ being the cardinality of $S$. Each *content-based filtering agent h* filters documents received from its respective *information agent h* by recalculating their relevance by means of a linguistic matching function

$$e_h : (S \times [-0.5, 0.5)) \times S \to S \times [-0.5, 0.5),$$

which is defined to model the semantics of threshold weights associated with the query terms. Different *content-based filtering agents* can have different threshold matching functions. For example, some linguistic matching functions that we can use are:

1. $e^1(\Delta(g\cdot F(d_j,t_i)), p_i^1) = \begin{cases} (s_g,0) & \text{if } \Delta(g\cdot F(d_j,t_i)) \geqslant (p_i^1,0) \\ (s_0,0) & \text{otherwise.} \end{cases}$
2. $e^2(\Delta(g\cdot F(d_j,t_i)), p_i^1) = \begin{cases} \Delta(g\cdot F(d_j,t_i)) & \text{if } \Delta(g\cdot F(d_j,t_i)) \geqslant (p_i^1,0) \\ (s_0,0) & \text{otherwise.} \end{cases}$
3. $e^3(\Delta(g\cdot F(d_j,t_i)), p_i^1) = \begin{cases} \Delta(min\{g, 0.5 + g\cdot F(d_j,t_i)\}) & \text{if } \Delta(g\cdot F(d_j,t_i)) \geqslant (p_i^1,0) \\ \Delta(max\{0, g\cdot F(d_j,t_i) - 0.5\}) & \text{otherwise.} \end{cases}$

Then, each *content-based filtering agent h* calculates a new set of relevance degrees $RN^h = \{rn_j^h, j=1,\dots,\#(D^h)\}$ characterizing the documents $D^h$, which is obtained as

$$rn_j^h = \bar{x}^e[e_h(\Delta(g\cdot F(d_j^h,t_1)), p_1^1),\dots,e_h(\Delta(g\cdot F(d_j^h,t_m)), p_m^1)]$$
$$= \Delta\left(\sum_{i=1}^{m} \frac{1}{m}\,\Delta^{-1}(e_h(\Delta(g\cdot F(d_j^h,t_i)), p_i^1))\right).$$

- *Step 8*: The *task agent* receives from every *content-based filtering agent h* a set of documents and their new relevance $(D^h, RN^h)$. It also receives a set of linguistic degree of satisfaction $C^h = \{c_1^h, c_2^h, \dots, c_m^h\}$, $c_i^h \in S\times[-0.5, 0.5)$ of $D^h$ with regard to every term of the query, which is calculated as

$$c_i^h = \bar{x}^e[e_h(\Delta(g\cdot F(d_1^h,t_i)), p_i^1),\dots,e_h(\Delta(g\cdot F(d_{\#(D^h)}^h,t_i)), p_i^1)]$$
$$= \Delta\left(\sum_{j=1}^{\#(D^h)} \frac{1}{\#(D^h)}\,\Delta^{-1}(e_h(\Delta(g\cdot F(d_j^h,t_i)), p_i^1))\right).$$

Then, the *task agent* selects the number of documents to be retrieved from each *content-based filtering agent h*. To do so, it applies the following three steps:

- *Step 8.1*: The *task agent* orders $D^h$ with respect to the new relevance $RN$.
- *Step 8.2*: The *task agent* aggregates through a 2-tuple linguistic weighted average operator, for example $\bar{x}_l^w$, both the satisfaction of the query terms $C^h$ and the relative importance weights that the user assigned to the query terms, $\{p_i^2, i=1,\dots,m\}$. In such a way, it obtains a satisfaction degree $\lambda^h \in S\times[-0.5, 0.5)$ for each *content-based filtering agent h*, which is computed as follows:

$$\lambda^h = \bar{x}_l{}^w[(c_1^h,(p_1^2,0)),\dots,(c_m^h,(p_m^2,0))].$$

- *Step 8.3*: To gather the better documents from *content-based filtering agents*, the *task agent* selects a number of documents $k(D^h)$ from every *content-based filtering agent h* being proportional to its respective degree of satisfaction $\lambda^h$ [6]:

$$k(D^h) = round\left(\frac{\sum_{i=1}^{n}\#(D^i)}{n}\cdot P_s^h\right),$$

where $P_s^h = \Delta^{-1}(\lambda^h)/\sum_{i=1}^{n} \Delta^{-1}(\lambda^h)$ is the probability of selection of the documents from *content-based filtering agent* $h$.

- *Step 9*: The *collaborative filtering agent* receives from the *task agent* a list of documents $DV = \{d_1^V, \ldots, d_v^V\}$ ordered with respect to their relevance $RV$, such that

  1. $r_j^V \geqslant r_{j+1}^V$,
  2. for a given document $d_j^V \in DV$ there exists a $h$ such that $d_j^V \in D^h$ and $r_j^V \in RN^h$, and
  3. $\#(DV) = v \leqslant \sum_{i=1}^{n} k(D^i)$.

Then, *collaborative filtering agent* filters the documents provided by the *task agent* using the recommendations on such documents provided by other users in previous searches which are stored in a *collaborative recommender system*. This is done in the following steps:

- *Step 9.1*: The *collaborative filtering agent* asks *collaborative recommender system* the recommendations existing on $DV$ associated with the information need category $\mathscr{A}_i$ expressed by the user and retrieves them,

$$RC^{\mathscr{A}_i} = \{RC_1^{\mathscr{A}_i}, \ldots, RC_v^{\mathscr{A}_i}\} RC_j^{\mathscr{A}_i} \in S \times [-0.5, 0.5).$$

- *Step 9.2*: The *collaborative filtering agent* filters the documents by recalculating their relevance using these recommendations $RC^{\mathscr{A}_i}$. Then, for each document $d_j^V \in DV$ a new linguistic relevance degree $r_j^{NV}$ is calculated from $r_j^V$ and $RC_j^{\mathscr{A}_i}$ by means of the 2-tuple weighted operator $\bar{x}^w$ defined in Definition 4

$$r_j^{NV} = \bar{x}^w(r_j^V, RC_j^{\mathscr{A}_i}), \text{ using for example the weighting vector } W = [0.6, 0.4].$$

- *Step 10*: The *interface agent* receives from the *collaborative filtering agent* a list of documents $DW = \{d_1^W, \ldots, d_w^W\}$ ordered with respect to their relevance $RW$, such that:

  1. $r_j^W \geqslant r_{j+1}^W$,
  2. for a given document $d_j^W \in DW$ there exists a $i$ such that $d_j^W = d_i^V$ and $r_j^W = r_i^{NV}$, and
  3. $\#(DW) = w \leqslant v = \#(DV)$.

Then, the *interface agent* filters these documents in order to give to the user only those documents that fulfill better his/her needs, which we call $Df$. For example, it can select a fixed number of documents $K$ and to show the $K$ best documents.

### 3.3.2. Feedback phase

This phase is related to the activity developed by the *collaborative recommender system* once user has taken some of documents retrieved by the multi-agent system.

In the collaborative recommender systems the people collaborate to help one another to perform filtering by recording their reactions to documents they read [19,27]. In a typical collaborative system people provide evaluation judgements or annotations on documents as inputs (feedback information), which the system then aggregates obtaining recommendations that later can be reused to assist another people in their search processes. In our multi-agent model this feedback activity is developed in the following steps (in Fig. 2 the discontinuous lines symbolize this phase):

- *Step 1*: The *interface agent* gives the user's identity $\mathscr{ID}$ (usually his/her e-mail) together with the set of documents $DU = \{d_1^U, \ldots, d_u^U\}$, $u \leqslant \#(Df)$ used by the user to the *collaborative filtering agent*.
- *Step 2*: The *collaborative filtering agent* asks user his/her opinion or evaluation judgements about $DU$, for example by means of an e-mail.
- *Step 3*: The *Internet user* communicates his/her linguistic evaluation judgements to the *collaborative recommender system*, $rc_y$, $y = 1, \ldots, \#(DU)$, $rc_y \in S$.
- *Step 4*: The *collaborative recommender system* recalculates the linguistic recommendations of set of documents $DU$ by aggregating again the opinions provided by other users together with those provided by the Internet user. This can be done using the 2-tuple aggregation operator $\bar{x}^{\mathrm{e}}$ given in Definition 3. Then, given a chosen document $d_y^U \in DU$ that receives a recommendation or evaluation judgement $rc_y$ from the Internet user, and supposing that in the collaborative recommender system there exists a set of stored linguistic recommendations $\{rc_1, \ldots, rc_M\}$, $rc_i \in S$ associated with $d_y^U$ for the information need category $\mathscr{A}_i$, which were provided by $M$ different users in previous searches, then a new value of recommendation of $d_y^U$ is obtained as

$$RC_y^{\mathscr{A}_i} = \bar{x}^{\mathrm{e}}[(rc_1, 0), \ldots, (rc_M, 0), (rc_y, 0)].$$

### 3.4. Analysis of the performance

In this section we analyze the performance of our new multi-agent model. To do so, on the one hand, we compare its structure and operation with respect to the model developed in [6] and study its main advantages and drawbacks, and on the other hand, we research some aspects that can contribute to improve its performance, e.g. the critical number of content-based filtering agents to use in the retrieval activity.

#### 3.4.1. Comparative study

As aforementioned, in this paper we propose a new multi-agent model in order to improve the retrieval activity of model defined in [6]. It is known that the more knowledge we have on user information needs the more possibilities we have to achieve our goal. Thus, we decide to incorporate two new elements in the model proposed in [6]: a new expression language to get a larger and better knowledge of user information needs and several technical elements that allow to exploit that knowledge to improve retrieval issues. These elements allow users to express their information needs by means of multi-weighted queries (each term can be weighted by two weights ) together with information need categories (in [6] users only used weighted queries by one weight to express their information needs) and incorporate several information filtering agents in the multi-agent model to exploit that knowledge (in [6] we did not apply any information filtering tool). Obviously, these new elements provide an additional value to the model defined in [6]. In what follows we analyze the main drawbacks and advantages of our proposal:

1. The structure of the new multi-agent model is more complicated. It contains two new action levels (the level of content-based filtering agents and the level of collaborative filtering agent) to

develop the information filtering activity. Therefore, the design of the information flows among agents is more complex and its implementation more difficult. However, this new structure allows to develop a more precise search on the Web.

2. The success of this model depends on the users' collaboration. The model provides users new mechanisms to participate in the information search process but this requires their collaboration. In the worst case, i.e. when the user does not use threshold weights, information need categories and does not collaborate in the generation of recommendations, the system obtains the same issues that the model defined in [6]. When users collaborate with the system in some of above aspects the success of the new system is guaranteed because it develops a more guided search.
3. The operation of this multi-agent model is also more complicated. It also presents an information retrieval phase as in [6] and a new phase that we call feedback phase. Therefore, the system carries out many more activities. However, we observe that these activities do not penalize seriously the respond time of the information search process due to the following reasons:
   - In a particular moment, when a user is searching information, the response time depends only on the retrieval phase. Thus, the feedback phase does not overload the response time of the information search process. We should point out that both phases are complementary with the use of system by the users, i.e., on the one hand, if users are satisfied with issues obtained in the retrieval phase developed by the system then the possibilities of their participation in the feedback phase increase, and, on the other hand, if users participate in the feedback phase then the success of retrieval phase also increases.
   - Apparently, the retrieval phase of this new model requires much more time than the retrieval phase of the model defined in [6] because it includes two new activities, the content-based filtering activity and the collaborative filtering activity. However, this is not the case. The first activity consists of recalculating the relevance degrees of retrieved documents for each information agent by applying a threshold matching function, which is an easy computation that does not need much time in each content-based filtering agent. On the other hand, the activity of collaborative filtering agent is developed in two steps. In the first one (Step 9.1 of the retrieval phase) the collaborative filtering agent asks collaborative recommender systems the recommendations existing on the retrieved document associated with the information need category expressed by the user, a communication that clearly can overload the response time of the system. However, the impact of such communication can be reduced if the Step 9.1 is applied simultaneously to the processing of the user multi-weighted query. This means retrieving all documents recommended in the information need category provided by the user from collaborative recommender system. Thus, when the collaborative filtering agent receives from task agent the relevant documents to the user query, it only has to recalculate relevance degrees using the recommendations previously retrieved. In such a way, Step 9.1 is developed quickly. In the second step (Step 9.2 of the retrieval phase) the relevance degrees of documents are recalculated by means of the computing process of two values and this is a quick and easy computation.
   - The rest of agents of the new model work similarly as in the model defined in [6], and therefore, they do not add more time to the information search process.
4. The query language of the multi-agent model provides users more expression possibilities to represent their information needs.

5. With such language and the filtering tools this new model improves the issues of the information search process of the model proposed in [6] because it can develop a more precise information search process.
6. Consequently, it has more possibilities to improve users' degree of satisfaction, although, as aforementioned, it requires their participation.

*3.4.2. Improving the performance of proposed multi-agent model*

The response time of the information search process of new model can be improved with the following small considerations:

1. As aforementioned, if in the implementation of the model the Step 9.1 is applied simultaneously to the processing of the user multi-weighted query we get that the retrieval of recommendations does not increase the response time of system.
2. If we reduce the processing time required to carry out the activity of level of content-based filtering agents then the response time of system can be reduced. There are two possibilities:
   (a) Reducing the number of content-based filtering agents that participate in the search process. For example, we can consider that the number of content-based filtering agents is limited by the number of threshold matching functions, and then, to distribute all information agents among considered content-based filtering agents.
   (b) In the implementation of model we can include the activity of each content-based filtering agent $h$ in its respective information agent $h$. This means substituting the relevance computation developed in the information agents by the relevance computation developed in the content-based filtering agents.

## 4. Example

In this section we present an example of the activity of new multi-agent model. For this purpose, we consider a view of a single user $I$, as it was set out in Fig. 3. Furthermore, we use the set of seven labels (i.e., $g = 6$), shown in Fig. 1, to represent the linguistic information.

Suppose that user $I$ expresses his/her information needs by the following linguistic multi-weighted query $[(Agents, VH, VH), (Web, H, M)]$ and the following information need category $\mathscr{A}_i = Web\ Mining$. With such request the user is expressing to have a stake in documents dealing with the topic *Agents* in a *Web* context at least in very high and high degrees respectively, i.e., by documents about *Web Agents* in a high degree, and in addition, he/she wants to analyze these documents from the perspective suggested by the topic "*Web Mining*. Furthermore, user prefers documents in which the topic "*Agents*" to be more important than *Web*, and this is expressed explicitly by assigning to these topics the linguistic relative importance weights $VH$ and $M$, respectively. Together with this request the user gives interface agent his/her identity (e-mail).

The interface agent transfers all above information to the collaborative filtering agent. This communicates the linguistic multi-weighted query to the task agent. The task agent gives the terms of the user query together with their respective linguistic threshold weights to the content-based filtering agent level. Each content-based filtering agent only passes the terms of the query to its respective information agent. The information agents search in the information source level those documents

Table 1
Sets of documents for the terms ‘agents’ and ‘web’

| $(D^h, R^h)$ | $d_j^h$ | $r_j^h$ |
|---|---|---|
| $(D^1, R^1)$ | http://phonebk.duke.edu/clients/tnfagent.html | (H, 0.2) |
| | http://webhound.www.media.mit.edu/projects/webhound/doc/Webhound.html | (H, 0.2) |
| | http://www.elet.polimi.it/section/compeng/air/agents/ | (H, −0.4) |
| | http://www.cs.bham.ac.uk/ámw/agents/links/ | (M, 0) |
| | http://groucho.gsfc.nasa.gov/Code_520/Code_522/Projects/Agents/ | (L, 0.4) |
| $(D^2, R^2)$ | http://lcs.www.media.mit.edu/people/lieber/Lieberary/Letizia/Letizia.html | (VH, 0.4) |
| | http://www.osf.org/ri/contracts/6.Rationale.frame.html | (M, 0.2) |
| | http://www.info.unicaen.fr/ serge/sma.html | (M, 0.2) |
| | http://www.cs.umbc.edu/ cikm/1994/iia/papers/jain.html | (L, 0.4) |
| | http://www.hinet.com/realty/edge/gallery.html | (VL, 0.2) |
| $(D^3, R^3)$ | http://activist.gpl,ibm.com/WhitePaper/ptc2.htm | (VH, 0.4) |
| | http://www.cs.umbc.edu/ cikm/iia/submitted/viewing/chen.html | (H, −0.4) |
| | http://www.psychology.nottingham.ac.uk:80/aigr/research/agents/agents.html | (H, −0.4) |
| | http://netq.rowland.org/isab/isab.html | (M, 0) |
| | http://maple.net/gbd/salagnts.html | (VL, −0.4) |
| $(D^4, R^4)$ | http://www.ncsa.uiuc.edu/SDG/IT94/Proceedings/Agents/spetka/spetka.html | (VH, 0.4) |
| | http://mmm.wiwi.hu-berlin.de/MMM/cebit_engl.html | (H, -0.4) |
| | http://foner.www.media.mit.edu/people/foner/Julia/subsection3_2_2.html | (L, 0.4) |
| | http://www.cs.bham.ac.uk/ mw/agents/index.html | (L, 0.4) |
| | http://www.ffly.com/html/About1.html | (VL, 0.2) |

related to the terms of the query, and get a list with the most relevant links. For instance, each information agent $h$ $(h = 1, \ldots, 4)$ may retrieve a set of five links, $D^h$ and their relevance $R^h$ (see Table 1). For example, supposing that $F(d_1^1, Agents) = 0.9$ and $F(d_1^1, Web) = 0.5$, then $r_1^1$ is obtained as

$$r_1^1 = \bar{x}^e[\Delta(6 \cdot F(d_1^1, Agents)), \Delta(6 \cdot F(d_1^1, Web))] = \Delta\left(6 \cdot \frac{0.9 + 0.5}{2}\right) = (H, 0.2).$$

Each information agent $h$ gives back to its respective content-based filtering agent $h$ a set of documents $D^h$ together with its relevance $R^h$ and its original representation with respect to the terms of query, i.e., $\{(F(d_j^h, Agents), F(d_j^h, Web)),\ j = 1, \ldots, 5\}$. Then, each content-based filtering agent $h$ filters the received documents by applying the threshold semantics by means of a linguistic matching function $e_h$ to recalculate the relevance. Suppose that the content-based filtering agents filter documents obtain the following new linguistic relevance degrees $RN^h$:

$$rn_1^1 = (M, -0.3), \quad rn_3^1 = (L, 0.4), \quad rn_3^1 = (M, 0), \quad rn_4^1 = (M, -0.3), \quad rn_5^1 = (N, 0),$$

$$rn_1^2 = (H, 0), \quad rn_2^2 = (H, -0.4), \quad rn_3^2 = (M, 0.3), \quad rn_4^2 = (VL, 0.3), \quad rn_5^2 = (L, 0.3),$$

$$rn_1^3 = (M, -0.3), \quad rn_2^3 = (N, 0), \quad rn_3^3 = (N, 0), \quad rn_4^3 = (M, 0), \quad rn_5^3 = (N, 0),$$

$$rn_1^4 = (M, 0.2), \quad rn_2^4 = (M, 0.4), \quad rn_3^4 = (L, 0.4), \quad rn_4^4 = (N, 0), \quad rn_5^4 = (VL, 0.1),$$

where, for example, if the content-based filtering agent 1 uses the linguistic matching function $e^2$ and the representation of the documents of $D^1$ is the following:

| | Agents | Web |
|---|---|---|
| $F(d_1^1, - - -)$ | 0.9 | 0.5 |
| $F(d_2^1, - - -)$ | 0.6 | 0.8 |
| $F(d_3^1, - - -)$ | 0.2 | 1 |
| $F(d_4^1, - - -)$ | 0.9 | 0.1 |
| $F(d_5^1, - - -)$ | 0.4 | 0.4 |

then, the new linguistic relevance degrees are obtained as

$$rn_1^1 = \bar{x}^e[e^2(\Delta(6 \cdot 0.9), VH), e^2(\Delta(6 \cdot 0.5), H)]$$
$$= \Delta\left(\tfrac{1}{2} \cdot (\Delta^{-1}(VH, 0.4) + \Delta^{-1}(N, 0))\right) = (M, -0.3),$$

$$rn_2^1 = \bar{x}^e[e^2(\Delta(6 \cdot 0.6), VH), e^2(\Delta(6 \cdot 0.8), H)]$$
$$= \Delta\left(\tfrac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(VH, -0.2))\right) = (L, 0.4),$$

$$rn_3^1 = \bar{x}^e[e^2(\Delta(6 \cdot 0.2), VH), e^2(\Delta(6 \cdot 1), H)]$$
$$= \Delta\left(\tfrac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(P, 0))\right) = (M, 0),$$

$$rn_4^1 = \bar{x}^e[e^2(\Delta(6 \cdot 0.9), VH), e^2(\Delta(6 \cdot 0.1), H)]$$
$$= \Delta\left(\tfrac{1}{2} \cdot (\Delta^{-1}(VH, 0.4) + \Delta^{-1}(N, 0))\right) = (M, -0.3),$$

$$rn_5^1 = \bar{x}^e[e^2(\Delta(6 \cdot 0.4), VH), e^2(\Delta(6 \cdot 0.4), H)]$$
$$= \Delta\left(\tfrac{1}{2} \cdot (\Delta^{-1}(N, 0) + \Delta^{-1}(N, 0))\right) = (N, 0).$$

As can be observed the application of the threshold weights can change the relevance of the documents, and therefore, the ordering among the documents. For example, after to recalculate the relevance, the third more relevant document provided by the information agent 1 is considered the best one by the content-based filtering agent 1.

In each content-based filtering agent is also calculated the satisfaction degrees of terms of query. Consider that the obtained linguistic satisfaction degrees are the following:

$$[c_1^1, c_2^1] = [(L, 0.16), (L, 0.16)], \quad [c_1^2, c_2^2] = [(H, 0.1), (H, 0.4)],$$

$$[c_1^3, c_2^3] = [(M, 0.1), (M, 0.3)], \quad [c_1^4, c_2^4] = [(M, 0.4), (L, 0.2)],$$

where, for example, in the first content-based filtering agent the first linguistic satisfaction degree is calculated as follows:

$$c_1^1 = \bar{x}^e[e(\Delta(6\cdot 0.9), VH), e(\Delta(6\cdot 0.6), VH), e(\Delta(6\cdot 0.2), VH), e(\Delta(6\cdot 0.9), VH),$$
$$e(\Delta(6\cdot 0.4), VH)]$$
$$= \Delta\left(\frac{1}{5}(\Delta^{-1}(VH,0.4) + \Delta^{-1}(N,0) + \Delta^{-1}(N,0) + \Delta^{-1}(VH,0.4) + \Delta^{-1}(N,0))\right)$$
$$= \Delta\left(\frac{10.8}{5}\right) = (L, 0.16).$$

Then, each content-based filtering agent $h$ sends task agent its sets ($D^h$, $RN^h$, $C^h$). In the task agent the documents in $D^h$ are ordered with respect to new relevance $RN^h$. Then, it calculates a global satisfaction degree of query for each content-based filtering agent $h$, called $\lambda^h$. This degree is calculated by aggregating both the satisfaction degrees $C^h$ and the relative importance degrees provided by the user $\{p_i^2,\ i=1,\dots,m\}$ by means of the 2-tuple linguistic weighted operator $\bar{x}_l{}^w$. In our example, we obtain the following satisfaction degrees:

$$\lambda^1 = (L, 0.16),\quad \lambda^2 = (H, 0.2125),\quad \lambda^3 = (M, 0.175),\quad \lambda^4 = (M, -0.05),$$

where for example, $\lambda^2$ is calculated as follows:

$$\lambda^2 = \bar{x}_l^w[((H,0.1),(VH,0)),((H,0.4),(M,0))] = \Delta\left(\frac{4.1\cdot 5 + 4.4\cdot 3}{8}\right) = (H, 0.2125).$$

In the next step, the task agent gathers the best documents from those provided by each content-based filtering agent $h$ according to its respective satisfaction degree $\lambda^h$. To do so, it calculates the probabilities of selection of the documents of each content-based filtering agent $h$ obtaining the following selection probabilities: $P_s^1 = 0.1728$, $P_s^2 = 0.3371$, $P_s^3 = 0.2541$, $P_s^4 = 0.2360$, where for example, $P_s^2$ is obtained as

$$P_s^2 = \frac{\Delta^{-1}(\lambda^2)}{\Delta^{-1}(\lambda^1) + \Delta^{-1}(\lambda^2) + \Delta^{-1}(\lambda^3) + \Delta^{-1}(\lambda^4)} = \frac{4.2125}{12.4975} = 0.3371.$$

With these selection probabilities the task agent calculates the number of documents $k(D^h)$ to select from each content-based filtering agent $h$

$$k(D^1) = 1,\quad k(D^2) = 2,\quad k(D^3) = 1,\quad k(D^4) = 1,$$

where for example, $k(D^2) = round((\sum_{i=1}^{4} \#(D^i)/4)\cdot P_s^2) = round(5\cdot 0.3371) = 2$. Hence, the list of documents $DV$ ordered by relevance $RV$ that the collaborative filtering agent receives from the task agent is the following:

$$(d_1^V, r_1^V) = (d_1^2, rn_1^2) = (d_1^2, (H,0)),\quad (d_2^V, r_2^V) = (d_2^2, rn_2^2) = (d_2^2, (H,-0.4)),$$
$$(d_3^V, r_3^V) = (d_2^4, rn_2^4) = (d_2^4, (M,0.4)),\quad (d_4^V, r_4^V) = (d_3^1, rn_3^1) = (d_3^1, (M,0)),$$
$$(d_5^V, r_5^V) = (d_4^3, rn_4^3) = (d_4^3, (M,0)).$$

Now, the collaborative filtering agent filters these documents by considering the recommendations on these documents proposed by other users. To do so, it recalculates again their relevance by including in the computation of the relevance the recommendations provided by the collaborative recommender system. Suppose that the recommendations existing on these documents in the collaborative recommender system are the following:

| Documents for topic $\mathscr{A}_i =$ "*web mining*" | User's judgements | Recommendation $RC_j^{\mathscr{A}_i}$ |
|---|---|---|
| $d_1^V$ | $(Id1, H), (Id4, M), (Id5, M)$ | $(M, 0.33)$ |
| $d_2^V$ | $(Id1, VH)$ | $(VH, 0)$ |
| $d_3^V$ | $(Id2, L), (Id3, VL)$ | $(L, -0.5)$ |
| $d_4^V$ | — | — |
| $d_5^V$ | $(Id1, H), (Id4, VH), (Id5, H)$ | $(H, 0.33)$ |

In the table we can observe that for the document $d_4^V$ there not exist user's judgements stored. And for example, the recommendation $RC_1^{\mathscr{A}_i}$ is obtained as

$$RC_1^{\mathscr{A}_i} = \bar{x}^{\mathrm{e}}[(H,0),(M,0),(M,0)] = \Delta\left(\frac{4+3+3}{3}\right) = (M, 0.33).$$

Then, using these recommendations the collaborative filtering agent recalculates the relevance of documents $DV$ by means of the 2-tuple weighted operator $\bar{x}^w$ with the weighting vector [0.6,0.4], obtaining the following new set of relevance degrees $RNV$:

$$r_1^{NV} = \bar{x}^w[(H,0),(M,0.33)] = \Delta(4\cdot 0.6 + 3.33\cdot 0.4) = \Delta(3.732) = (H, -0.268),$$

$$r_2^{NV} = \bar{x}^w[(H,-0.4),(VH,0)] = \Delta(3.6\cdot 0.6 + 5\cdot 0.4) = \Delta(4.16) = (H, 0.16),$$

$$r_3^{NV} = \bar{x}^w[(M,0.4),(L,-0.5)] = \Delta(3.4\cdot 0.6 + 1.5\cdot 0.4) = \Delta(2.64) = (M, -0.36),$$

$$r_4^{NV} = (M,0) = r_4^V (\textit{This relevance value does not change}),$$

$$r_5^{NV} = \bar{x}^w[(M,0),(H,0.33)] = \Delta(3\cdot 0.6 + 4.33\cdot 0.4) = \Delta(3.532) = (H, -0.468).$$

Hence, the list of documents $DW$ ordered by relevance $RW$ that the interface agent receives from the collaborative filtering agent is the following:

$$(d_1^W, r_1^W) = (d_2^V, r_2^{NV}) = (d_2^V, (H, 0.16)), \quad (d_2^W, r_2^W) = (d_1^V, r_1^{NV}) = (d_1^V, (H, -0.268)),$$
$$(d_3^W, r_3^W) = (d_5^V, r_5^{NV}) = (d_5^V, (H, -0.468)), \quad (d_4^W, r_4^W) = (d_4^V, r_4^{NV}) = (d_4^V, (M, 0)),$$
$$(d_5^W, r_5^W) = (d_3^V, r_3^{NV}) = (d_3^V, (M, -0.36)).$$

In the last step of the algorithm, the interface agent filters this final ranked list of documents and gives to the internet user the most relevant documents $(Df, Rf)$. For example if the fixed number of documents in the interface agent is $K=3$ then the system shows the following documents:

$$(d_1^f, r_1^f) = (d_2^2, r_1^W) = (http://www.osf.org/ri/contracts/6.Rationale.frame.html, (H, 0.16)).$$

$$(d_2^f, r_2^f) = (d_1^2, r_2^W) = (http://lcs.www.media.mit.edu/people/lieber/Lieberary/Letizia/Letizia.html, (H, -0.268)),$$

$$(d_3^f, r_3^f) = (d_4^3, r_3^W) = (http://netq.rowland.org/isab/isab.html, \ (H, -0.468)).$$

Later, the multi-agent system has to carry out the feedback activity in which the internet user is asked by his/her opinion about shown documents that he/she has used. This activity is easily done, and when user provides his/her evaluation judgements, the collaborative recommender system stores them and recalculates the recommendations for those documents using the operator $\bar{x}^e$ as was done above.

## 5. Concluding remarks

We have presented a new fuzzy linguistic multi-agent model based on linguistic 2-tuple representation that incorporates in its activity the two more important information filtering techniques: content-based filtering and collaborative filtering. In such a way, we improve the search processes on the Web and increase the users' satisfaction degrees.

In the future, we want to study proposals that allow users to express better both their information needs and their evaluation judgements on documents to generate the recommendations.

## Acknowledgements

This work has been partially supported by Research Projects TIC2002-03348 and TIC2002-03276.

## References

[1] C. Basu, H. Hirsh, W. Cohen, Recommendation as classification: using social and content-based information in recommendation, Proc. 15th Nat. Conf. on Artificial Intelligence, 1998, pp. 714–720.

[2] G. Boone, Concept features in RE:agent, an intelligent email agent, Proc. Autonomous Agents, 1998, pp. 141–148.

[3] W. Brenner, R. Zarnekow, H. Witting, Intelligent Software Agent, Foundations and Applications, Springer, Berlin, Heidelberg, 1998.

[4] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, M. Sartin, Combining content-based and collaborative filters in an online newpaper, Proc. ACM SIGIR'99 Workshop on Recommender Systems-Implementation and Evaluation, Berkeley, CA, 1999.

[5] M. Chau, D. Zeng, H. Chen, M. Huang, D. Hendriawan, Design and evaluation of a multi-agent collaborative Web mining system, Decision Support Systems 35 (2003) 167–183.
[6] M. Delgado, F. Herrera, E. Herrera-Viedma, M.J. Martín-Bautista, L. Martínez, M.A. Vila, A communication model based on the 2-tuple fuzzy linguistic representation for a distributed intelligent agent system on Internet, Soft Comput. 6 (2002) 320–328.
[7] M. Delgado, F. Herrera, E. Herrera-Viedma, M.J. Martín-Bautista, M.A. Vila, Combining linguistic information in a distributed intelligent agent model for information gathering on the Internet, in: P.P. Wang (Ed.), Computing with Words, Wiley, New York, 2001, pp. 251–276.
[8] B. Fazlollahi, R.M. Vahidov, R.A. Aliev, Multi-agent distributed intelligent system based on fuzzy decision making, Internat. J. Intelligent Systems 15 (2000) 849–858.
[9] J. Ferber, Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence, Addison-Wesley Longman, New York, 1999.
[10] N. Good, J.B. Shafer, J.A. Konstan, A. Borchers, B.M. Sarwar, J.L. Herlocker, J. Riedl, Combining collaborative filtering with personal agents for better recommendations, Proc. 16th National Conf. on Artificial Intelligence, 1999, pp. 439–446.
[11] F. Herrera, E. Herrera-Viedma, Aggregation operators for linguistic weighted information, IEEE Trans. Systems, Man Cybernet., Part A: Systems and Humans 27 (5) (1997) 646–656.
[12] F. Herrera, E. Herrera-Viedma, L. Martínez, C. Porcel, Information gathering on the internet using a distributed intelligent agent model with multi-granular linguistic information, in: V. Loia (Ed.), Fuzzy Logic and The Internet, Physica-Verlag, Springer, Wurzburg, Berlin, 2003, in press.
[13] F. Herrera, E. Herrera-Viedma, J.L. Verdegay, Direct approach processes in group decision making using linguistic OWA operators, Fuzzy Sets and Systems 79 (1996) 175–190.
[14] F. Herrera, L. Martínez, A 2-tuple fuzzy linguistic representation model for computing with words, IEEE Trans. Fuzzy Systems 8 (6) (2000) 746–752.
[15] F. Herrera, L. Martínez, A model based on linguistic 2-tuples for dealing with multigranularity hierarchical linguistic contexts in multiexpert decision-making, IEEE Trans. Systems, Man Cybernet., Part B: Cybernet. 31 (2) (2001) 227–234.
[16] F. Herrera, L. Martínez, The 2-tuple linguistic computational model. Advantages of its linguistic description, accuracy and consistency, Internat. J. Uncertainty , Fuzziness Knowledge-Based Systems 9 (2001) 33–48.
[17] E. Herrera-Viedma, Modeling the retrieval process of an information retrieval system using an ordinal fuzzy linguistic approach, J. Amer. Soc. Inform. Sci. Technol. 52 (6) (2001) 460–475.
[18] E. Herrera-Viedma, An information retrieval system with ordinal linguistic weighted queries based on two weighting elements, Internat. J. Uncertainty, Fuzziness Knowledge-Based Systems 9 (2001) 77–88.
[19] E. Herrera-Viedma, E. Peis, Evaluating the informative quality of documents in SGML-format using fuzzy linguistic techniques based on computing with words, Inform. Process. Manag. 39 (2) (2003) 195–213.
[20] N. Jennings, K. Sycara, M. Wooldridge, A roadmap of agent research and development, Autonom. Agents Multi-Agents Systems 1 (1998) 7–38.
[21] M. Kobayashi, K. Takeda, Information retrieval on the web, ACM Comput. Surveys 32 (2) (2000) 144–173.
[22] S. Lawrence, C. Giles, Searching the web: general and scientific information access, IEEE Comm. Mag. 37 (1) (1998) 116–122.
[23] H. Lieberman, Personal assistants for the Web: a MIT perspective, in: M. Klusch (Ed.), Intelligent Information Agents, Springer, Berlin, 1999, pp. 279–292.
[24] P. Maes, Agents that reduce work and information overload, Comm. ACM 37 (1994) 31–40.
[25] A. Moukas, G. Zacharia, P. Maes, Amalthaea and Histos: multiagent systems for WWW sites and representation recommendations, in: M. Klusch (Ed.), Intelligent Information Agents, Springer, Berlin, 1999, pp. 293–322.
[26] A. Popescul, L.H. Ungar, D.M. Pennock, S. Lawrence, Probabilistic models for unified collaborative and content-based recommendation in sparce-data environments. in: Proc. 17th Conf. on Uncertainty in Artificial Intelligence (UAI), San Francisco, 2001, pp. 437–444.
[27] P. Reisnick, H.R. Varian, Recommender Systems, Special issue of Comm. ACM 40 (3) (1997) 56–58.
[28] G. Salton, M.G. McGill, Introduction to Modern Information Retrieval, McGraw-Hill, New York, 1983.
[29] J.B. Schafer, J.A. Konstan, J. Riedl, E-Commerce recommendation applications, Data Mining Knowledge Discovery 5 (1/2) (2001) 115–153.

[30] K. Sycara, A. Pannu, M. Williamson, D. Zeng, Distributed intelligent agents, IEEE Expert (1996) 36–46.
[31] R.R. Yager, Protocol for negotiations among multiple intelligent agents, in: J. Kacprzyk, H. Nurmi, M. Fedrizzi (Eds.), Consensus Under Fuzziness, Kluwer Academic Publishers, Dordrecht, 1996, pp. 165–174.
[32] R.R. Yager, Intelligent agents for World Wide Web advertising decisions, Internat. J. Intelligent Systems 12 (1997) 379–390.
[33] R.R. Yager, Fusion of multi-agent preference orderings, Fuzzy Sets and Systems 112 (2001) 1–12.
[34] L.A. Zadeh, The concept of a linguistic variable and its applications to approximate reasoning. Part I, Inform. Sci. 8 (1975) 199–249;
L.A. Zadeh, The concept of a linguistic variable and its applications to approximate reasoning. Part II, Inform. Sci. 8 (1975) 301–357;
L.A. Zadeh, The concept of a linguistic variable and its applications to approximate reasoning. Part III, Inform. Sci. 9 (1975) 43–80.