A Fuzzy Linguistic Multi-agent Model Based on Semantic Web Technologies and User Profiles

Enrique Herrera-Viedma[1], Eduardo Peis[2], and José M. Morales-del-Castillo[2]

[1] Dpt. Computer Science and AI. University of Granada
viedma@decsai.ugr.es
[2] Dpt. Library and Information Science. University of Granada
epeis@ugr.es, josemdc@ugr.es

Summary. In this paper we define a model of a web multi-agent system that combines the use of Semantic Web technologies (like RDF and OWL languages to describe resources and define ontologies respectively) together with the application of user profiles to carry out its activity. It is designed using fuzzy linguistic techniques which allow dealing with information in a user-friendly way. The system activity is developed in two phases: retrieval phase to gather the documents from the Web, and feedback phase to update the elements used to filter the documents, i.e., user profiles and recommendation values. We focus on the analysis of the feedback phase. With this multi-agent system model the filtering possibilities are increased and, consequently, also the retrieval capabilities on the Web.

1 Introduction

The Internet harbours a vast amount of information that does not cease to grow exponentially. To avoid arriving to a situation of collapse, the actual model of Web needs to incorporate tools capable to handle and manage efficiently this huge quantity of resources. The solutions that exist nowadays are shown little efficient since they oblige users to lose great part of their time in deciding which documents are relevant and which not. Therefore it is becoming necessary to develop systems for searching and mining the Web that permit to improve the access to the information in an efficient way. At this moment, some of the more recurrent technologies to face this problem deal with the development of intelligent software agents [14, 27], the application of techniques of information filtering [37], and the development of the Semantic Web project [3, 4, 5].

Software agents applied to a Web-based environment are organized in distributed architectures [7, 13, 14, 26, 29] to mainly perform tasks of intermediation between users and the Web, e.g. assisting users in the information

E. Herrera-Viedma et al.: *A Fuzzy Linguistic Multi-agent Model Based on Semantic Web Technologies and User Profiles*, StudFuzz **197**, 105–120 (2006)
www.springerlink.com

retrieval process [7, 14, 26, 39]. These agents are entities capable to act in an autonomous way, processing and exchanging results with other agents [20], but need to have certain amount of knowledge about the system and the user to be able to develop these tasks. Having an extensive knowledge base about the user implies that the agent has access to an exhaustive and "first hand" information, but the user has to pay an expensive cost, being exposed to continuous consultations to maintain his/her information updated. An alternative solution is to provide the agent with a smaller knowledge base but supported on web ontologies [19, 20], forcing this way the agent to adopt an active stance and seek information about the user and the tasks that carries out in his/her interaction with the system. Agents are able to use different methods to get this information, e.g. by direct observation and imitation of users' behaviour (search strategies used, links visited during a session, etc.) or taking advantage of users' feedback [27].

Nevertheless the main problem of using agents is to find a flexible and agile communication protocol for exchanging information among agents, and between users and agents because of the great variety of forms the information is represented in the Web. A possible option that permits to reduce these agent-agent and user-agent communication problems is to apply fuzzy linguistic techniques that allow operating with the information by means of the use of linguistic labels [40]. The application of this flexible system of representation enables us to handle information with several degrees of truth, solving the problem of quantifying qualitative concepts. Some examples of the use of fuzzy linguistic tools in the design of multi-agent systems can be found in [10, 11, 22, 25].

On the other hand, information filtering techniques ease users the task of sorting out relevant documents from those that are not, thanks to the prior selection the system carries out of the resources that better fit users' needs, requirements and preferences. These needs, requirements and preferences are mostly defined in a normalized way in the form of user profiles [6]. The efficacy of these user profiles depends on the up-to-dateness of the information contained, so there should exist a mechanism capable to dynamically bring up to date this information and to reflect in "real time" the variations on users' behaviour in their interaction with the system. In [25] we proposed a fuzzy linguistic multi-agent system based on collaborative filtering tools but no characterization of user's preferences was used. As it is known, the problem is that the performance of these tools tend to fail when little is known about users. Therefore, the use of the user profiles could contribute to improve its activity.

Another possibility to improve the activity of a multi-agent system could be the use of some of the technologies of the Semantic Web project [3, 4, 5]. These technologies can be exploited to develop ontology-based infrastructures where agents can operate at semantic level with resources described using languages as XML (eXtensible Mark-up Language) or RDF (Resource Description Framework) [12, 30, 34, 35, 38] in a manner both interpretable by humans and machines. This common syntactic framework allows us to define an unique communication vocabulary among agents that could also be used to characterize the knowledge base of the system and even the semantics of resources or user profiles. In [41], for example, the use of the ontological language OWL (Ontology Web Language) as content language to exchange messages allows to enlarge the interoperability among agents (as a result of using a common convention for representing knowledge) and also to endow them with greater capacities for inference reasoning over resources.

The aim of this paper is to present a new model of fuzzy linguistic multi-agent system that involves the use of the Semantic Web technologies and user profiles dynamically updated to overcome the problems of the system presented in [25]. The Semantic Web technologies are used to endow the agents with a common communication language, to develop the ontologies of the system and to characterize resources and user profiles in a standardized way using RDF. We propose a method to automatically update the user profiles in function of the appraisals users provide about system's output. This updating method is designed using linguistic matching functions [23]. As in [25], the system activity presents two phases, retrieval and feedback. In this paper we focus on the latter.

The paper is structured as follows. Section 2 reviews the methodological tools employed in this research: the fuzzy linguistic tools, the filtering tools, and the Semantic Web technologies. Section 3 presents the new multi-agent model. Finally, some concluding remarks are pointed out in Section 4.

2 Methodological Tools

In this section, we present the tools that we apply to design our fuzzy linguistic multi-agent model.

2.1 Fuzzy Linguistic Tools

In the situations in which the information cannot be evaluated of quantitative form, but qualitative, it is necessary to apply a linguistic method. For example, when we try to qualify a phenomenon related to human perception, we use words in natural language instead of numerical values.

The *fuzzy linguistic approach* [40] and in particular, the *ordinal fuzzy linguistic approach* [21, 23, 24] are approximate techniques appropriate to deal with qualitative aspects of problems. An ordinal fuzzy linguistic approach is defined by considering a finite and totally ordered label set in the usual sense

$$S = \{s_i, i \in H = \{0, \ldots, T\}\}$$

and with odd cardinality (7 or 9 labels). The mid term representing an assessment of "approximately" 0.5 and the rest of the terms being placed symmetrically around it. The semantics of the linguistic term set is established from the ordered structure of the term set by considering that each linguistic term for the pair (s_i, s_{T-i}) is equally informative. For each label s_i is given a fuzzy number defined on the [0,1] interval, which is described by a linear trapezoidal membership function represented by the 4-tuple (a_i, b_i, α_i, β_i) (the first two parameters indicate the interval in which the membership value is 1.0; the third and fourth parameters indicate the left and right widths of the distribution). Furthermore, we require the following properties:

1. $-$ *The set is ordered*: $s_i \geqslant s_j$ *if* $i \geqslant j$.
2. $-$ *There is the negation operator*: $Neg(s_i) = s_j$, *with* $j = T - i$.
3. $-$ *Maximization operator*: $MAX(s_i, s_j) = s_i$ *if* $s_i \geqslant s_j$.
4. $-$ *Minimization operator*: $MIN(s_i, s_j) = s_i$ *if* $s_i \leqslant s_j$.

Additionally, we need aggregation operators to combine the linguistic information. In [22] is defined a useful operator, the Linguistic Ordered Weighted Averaging (LOWA) operator, which has been satisfactorily applied in different fields [21, 24, 25].

2.2 Filtering Techniques

Information filtering techniques deal with 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 resources available on the Web (usually, stored in HTML or XML documents) to assist people in their search processes [32], in most cases through the use of filtering agents [33]. Traditionally, these systems have fallen into two main categories [31]. *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 has short expressive capabilities. A possible solution to this problem is using user profiles as a tool for characterising users through explicit and implicit inputs that define both personal and professional information. In fact the integration of user profiles in the filtering process renders a powerful tool that allows fast and efficient filtering [37]. On the other hand, *collaborative filtering systems* use the information provided by many users to filter and recommend documents to a given user, ignoring the representation of documents. It's very usual to group the users in specific categories or stereotypes that are characterized by a set of default preferences and rules that represent the information needs and common search habits of a group of related users. These recommender systems tend to fail when little is known about a user, or when he/she has uncommon interests [31]. Several researchers are exploring hybrid content-based and collaborative recommender systems to smooth out the disadvantages of each one of them [1, 9, 15, 31, 37].

2.3 Semantic Web Technologies

The Semantic Web is an extension of the present Web, in which the information is gifted of a well defined meaning, permitting a better cooperation between humans and machines [3, 4, 5]. It is based on two main ideas: the "semantic" mark up of resources and the development of "intelligent" software agents capable to understand and to operate with these resources at semantic level [3,19].

The semantic backbone of the model is RDF/XML [2], a language that provides the necessary infrastructure to codify, exchange and reuse structured metadata in order to make them directly interpretable by machines. RDF/XML structures the information in assertions (resource-property-value triples), and uniquely identifies resources by means of URI's (Universal Resource Identifier), allowing intelligent software agents the knowledge extraction from and inference reasoning over resources (such as documents, user profiles or even queries) using ontologies.

Ontologies are defined in the Semantic Web context as a collection of relevant concepts of the knowledge shared by the members of a specific domain, the relations these concepts establish between them, and the axioms defined upon these concepts and relations [8, 16, 17]. Several ontology languages can be used for the design of web-based ontologies like OWL [28], a language with a great expressive capacity that allow us, for example, to define ontologies maintaining the RDF/XML syntactic convention or even to use it as common vocabulary for exchanging messages among agents.

3 The Fuzzy Linguistic Multi-agent Model Based on Semantic Web and User Profiles

In [25] we defined a model of a fuzzy linguistic multi-agent system to gather information on the Web that presents a hierarchical architecture composed of seven action levels: *internet users, interface agent, collaborative filtering agent, task agent, content-based filtering agent, information agents* and *information sources*. The main novelty of this model is the introduction of collaborative filtering agents in its architecture in order to increase its information filtering capabilities on the Web. Then, it develops its activity in two phases:

- *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.
- *Feedback phase:* This second phase coincides with the updating process of collaborative recommendations on desired documents existing in a 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.

The main drawback of this model is that it does not utilize user profiles to characterize the user preferences and this limits its performance possibilities.

To overcome the limitations of the model presented in [25] we define a new and enhanced model of fuzzy linguistic multi-agent system that improves information retrieval by means of the application Semantic Web technologies to set a base for the operation of software agents and user profiles to enrich the filtering activity.

This model presents a hierarchical structure with six action levels (*internet users, interface agent, filtering agent, task agent, information access,* and *information bases*), also two main activity phases, and a set of agents (*interface agent, filtering agent, task agent, profile agent, recommendation agent, information agents*) that work depending on the phase activity (see Fig. 1):

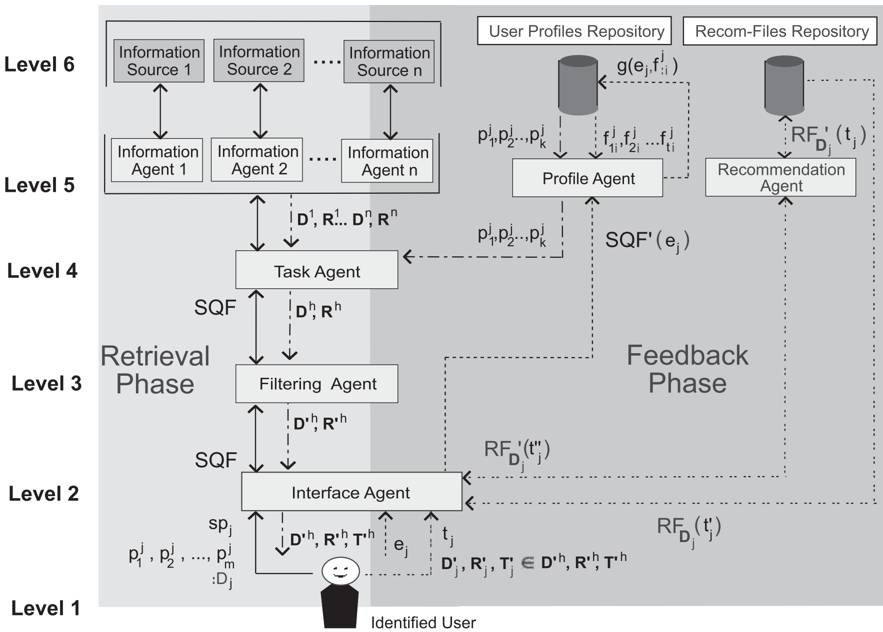

Fig. 1. Model architecture: Levels of action and processes.

- **Semantic Retrieval Phase**: This phase is similar to that developed by the model presented in [25], therefore, it coincides with the information gathering process developed by the multi-agent model itself. The main novelty of this phase is that the query language used is not a Boolean one, as in [25], but a semantic query language. The semantic information retrieval differs from the traditional information retrieval in the use of semantic query languages [18, 36], capable of comparing both literal and semantics structures. In such a way it is possible to obtain more accurate and contextualized answers to queries. The entities that participate in this phase are the following: *internet users (level 1), interface agent (level 2), filtering agent (level 3), task agent (level 4), information agent* as entity of information access *(level 5),* and *information sources* as the information bases that store the documents *(level 6).*

- **Feedback Phase**: This phase coincides with the updating process of both user profiles and collaborative recommendations on desired documents. Therefore, it involves two processes: "profile updating" process and "recommendation" process. Both processes need users' appraisals. In the first, they're given to assess the quality of the global answer provided by the system and, in the second one, to assess the retrieved documents. The "profile updating process" consists on the dynamic updating of the user profiles on the basis of the satisfaction degree the user expresses regard to the global results provided by the multi-agent system. In this process the entities that participate are the following: *internet users (level 1), interface agent (level 2), profile agent* as *entity of information access (level 5),* and the *user profiles repository* as *information bases* that store users' preferences *(level 6).* The "recommendation" process is similar to that defined in [25], therefore, it allows the user to express his/her opinions about any retrieved documents and using them the system can recalculate and update the stored recommendations about the desired documents. In this process the entities that participate are the following: *internet users (level 1), interface agent (level 2), recommendation agent* as *entity of information access (level 5),* and the *recommendation files repository* as *information bases (level 6).*

In the following subsection we analyze in detail the two action processes of the feedback phase.

3.1 Feedback Phase: User Profile Updating Process

Before to explain this process we have to clarify the inputs. This process works on three inputs:

1. *Semantic Query File (SQF):* This file is generated when the user formulates its requirements through the interface agent. This agent stores the user's $\mathrm{I}D_j$, the search parameters (sp_j) that define his/her "semantic" query and a set of preferences $\{p_1^j, p_2^j, \ldots, p_m^j\}$ on a file in RDF format we call *Semantic Query File* (SQF). These preferences for the user j are $\{p_1^j, p_2^j, \ldots, p_k^j\}$, where $0 \leqslant k \leqslant m$, being m the number of properties used to define a user profile and $p_i^j \in F_i$, being F_i the expression domain associated to the property i. For example, the user could provide his/her preferences about any of these four categories ($m = 4$) of basic preferences:

- *Document Type*: This preference defines the type of document the user prefers to retrieve among the ones the system provides. For example, in this case $F_1 = \{$*SciArticle, Proceedings, BookChapter, all*$\}$.
- *Search Context:* It consists of general topic categories that represent the main areas of the system domain. For example, if the domain of work of our system is "information systems" we could define $F_2 = \{$*decision support systems, information retrieval systems, geographical information systems, data mining, knowledge representation. all*$\}$.
- *Search aim:* It defines those tasks the user is going to carry out with the information to be retrieved. Depending on the nature of these tasks the system is able to decide which resources should be retrieved and which should not. In this case we could define different task categories such as $F_3 = \{$*research, teaching_bachelor, teaching_master, teaching_doctorate, studies_bachelor, studies_master, studies_doctorate, all*$\}$.
- *Date:* It refers to the updating or publication date of the resources to be retrieved. A set of different time intervals are defined to cover a wide range of values that vary from few months to several years (for exhaustive searches). For example, $F_4 = \{$*3 months, 6 months, 1 year, 3 years, 5 years, 10 years*$\}$.

Then, in this case, an user could provide from none ($k = 0$) up to four ($k = 4$) possible values of preference. Usually he/her will give us just one or two, being the other values taken from his/her stored profile.

An example of a SQF is shown in Fig. 2.

2. *User profiles*: We assume a repository that stores the user profiles in the system. Each user profile is determined by the user's identity and characterized by the particular values that each user has assigned in the categories of basic preferences. Each possible preference of the user profile has associated a linguistic frequency property (tagged as <*freq*>) representing how often a specific value has been used in satisfactory search processes. Thus, if $F_i = \{d_{1i}, d_{2i}, \ldots, d_{ti}\}$ then we define $f_i^j = \{f_{1i}^j, f_{2i}^j, \ldots, f_{ti}^j\}$ as the set of frequency values associated with each possible value $l \in \{1, \ldots, t\}$ of the property i in the profile of the user j. The range of possible values for the frequencies is defined in a set of seven linguistic labels, $S = \{$*always, almostAlways, mostTimes, sometimes, aFewTimes, almostNever, never*$\}$, i.e., $f_{li}^j \in S$. The following example (Fig. 3) shows how can be represented the frequency of use of a specific preference in a user profile.
3. *Global satisfaction degree:* This input is a key element in the profiles updating process, allowing the elicitation of users' tacit preferences. Once the user has checked out the resources retrieved by the system, he/she is required to assess the results providing a linguistic satisfaction degree e_j, that give us an approximate idea of the "kindness" of the preference values selected to define a query. In such a way, the system will be able to select (when needed) those preference values that have provided more satisfying answers to a user, instead of those values that have been more often selected. This global satisfaction degree can be defined using a set of seven

```
<?xml version="1.0" encoding="iso-8859-1"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns="http://www.ugr.es/~kishimaru/sqf#"
  xml:base="http://www.ugr.es/~kishimaru/sqf">
  <Query rdf:ID="query384">
    <user_ID>user-022005</user_ID>
    <appraisal></appraisal>
    <preferences_e>
      <preferences rdf:ID=”pref_384”>
        <docType>SciArticle</docType>
        <context>User_Modeling</context>
        <aim>Research_Article</aim>
        <date>3months</date>
        <value>NULL</value>
      </preferences>
    </preferences_e>
    <search_parameters_e>
     ....
    </search_parameters_e>
  </Query>
</rdf:RDF>
```

Fig. 2. Semantic Query File (SQF).

```
  ...
  <DocType rdf:ID="docType-pr001">
     <type1_e>
        <Type1 rdf:ID="type1-pr001">
           <type>SciArticle</type>
           <freq>AlmostAlways</freq>
        </Type1>
     </type1_e>
     <type2_e>
        <Type2 rdf:ID="type2-pr001">
           <type>Proceedings</type>
           <freq>AlmostNever</freq>
        </Type2>
     </type2_e>
  ...
  </DocType>
  ...
```

Fig. 3. Representation of preferences in a user profile.

linguistic labels $S' = \{$*Total, veryHigh, high, medium, low, veryLow, Null*$\}$, i.e., $e_j \in S'$.

Assuming the above inputs the profile updating process is developed in the following steps:

- **Step 1:** Once the user has checked the set of resources retrieved (D'^h, R'^h), the interface agent asks the user to express his/her satisfaction degree ($e_j \in S'$) with respect to the overall performance of the system, and it is stored in the semantic query file SQF.
- **Step 2:** The file SQF is transferred from the interface agent to the profile agent.
- **Step 3:** The profile agent carries out the updating of the linguistic frequency of use $\{f^j_{1i}, f^j_{2i}, \ldots, f^j_{ti}\}$ of each preference i defined in the user profile. To do that, we propose to use some type of matching function similar to the matching functions defined in the information retrieval systems to model the weighted queries. In particular, we use linguistic matching functions to model threshold weights in weighted user queries [23]. Assuming the linguistic satisfaction degree $e_j \in S'$ expressed by a user, if the property $i$is used by the user with a value l, then its respective associated frequency $f^j_{li} \in S$ is updated by means of the following linguistic matching function $g : S' \times S \to S$:

$$g\left(e_j, f^j_{li}\right) = \begin{cases} s_{Min\{a+\beta,T\}} & if \quad s_a \leqslant s_b \\ s_{Max\{0,a-\beta\}} & if \quad s_b < s_a \end{cases}$$

$$s_a, s_b \in S \,|a,\ b\ \in\ H = \{0, \ldots, T\}$$

such that, (i) $s_a = f^j_{li}$; (ii) $s_b = e_j$; and iii) β is a bonus value that rewards/penalizes the frequencies of the preferences of the user profile, which can be defined depending on the closeness between f^j_{li} and e_j for example, $\beta = round(2|b - a|/T)$. We should point out that this function is a non-decreasing matching function as the traditional threshold matching functions.

3.2 Feedback Phase: Recommendation Process

This process needs two basic inputs:

1. *Recommendation Files:* We assume the system has access to a repository of recommendation files (RF). On each document accessed in some moment of its history in the system there exists an associated recommendation file in RDF format (see Fig. 4) where is contained information about all the appraisals made by users that have read it formerly. This document-RF relation must be defined in the proper ontology to allow agents operating with both resources (this will allow, for example, showing users a list of titles of documents together with their respective recommendation values

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns="http://perso.wanadoo.es/kishimaru/recomFiles#"
xml:base="http://perso.wanadoo.es/kishimaru/recomFiles"
>
   <RecomFile rdf:ID="recomf001">
     <doc_ref>doc-pr008</doc_ref>
     <recom_value>High</recom_value>
     <recom_history>
        <R_history rdf:ID="histf001">
          <item>
            <RecomItem rdf:ID="form01-pr001">
              <appraissal>VeryHigh</appraissal>
              <topic>Data mining</topic>
              <user_ID>user-pr022005</user_ID>
            </RecomItem>
          </item>
          ....
        </R_history>
      </recom_history>
   </RecomFile>
</rdf:RDF>
```


Fig. 4. Representation of a Recommendation File (RF).

even though these data are located in different files). In a RF appears the ID of the document it belongs to, the current recommendation value and a set of log items containing previous appraisals about that document. Each log item is defined by an user's ID, his/her corresponding appraisal and the search context used in the query formulated by the user to retrieve that document (see "Search Context preference" in Section 3.1). This representation enables the adoption of different recommendation policies, allowing us, for example, to recalculate recommendation values based on the opinion of all the users, or just of some of them (e.g., using the appraisals given by those users who looked for information by the same topic). The decision to make a separate file defining the recommendations is justified by the dynamic nature of the recommendations. In such a way, we can modify the RF as many times as we like leaving untouched the representation of its associated document.

– *Recommendation value:* When a user checks out a document the systems asks him/her to recommend it to the rest of users of the system according to his/her opinion about the resource. In this way, each single person can help other users in the system to decide which resources are worthy to be read. In other words, the system enables its users to take advantage of the experience and knowledge of each others. The recommendation value (t_j)

provided by user j can be defined using a set of five linguistic labels $S'' = \{veryHigh, high, medium, low, veryLow\}$, i.e. $t_j \in S''$.

Then, the recommendation process is carried out in three steps:

- **Step 1:** Any document $D'_j \in (D'^h, R'^h, T'^h)$ has an associated relevance degree R'_j, and an associated set of "historical" recommendations $T'_j = \{t'_{1j}, t'_{2j}, \ldots, t'_{mj}\}$, being m any positive integer. When the user checks out the document D'_j is asked by the interface agent to appraise its global quality using a linguistic label $t_j \in S''$.
- **Step 2:** This appraisal is added to the RF associated to $D'_j(RF_{D'j})$ together with the "Search Context" preference value and the user's ID (stored in the SQF generated in the current search session), thus generating a new log item in the set of "historical" recommendation values.

In such a way, in a future search process the recommendation agent proceeds to dynamically recalculate the recommendation value of the document by means of an aggregation function, such as the LOWA operator [21, 22], that combines the whole set of historical recommendation values to calculate a new recommendation value (t''_j). This new value is stored in the <*recom_value*> tag of the $RF_{D'j}$, replacing the old t'_j, and can be used to be displayed in the list of retrieved resources, coupled with its respective document link.

4 Example

Suppose the following framework. John Quest, a researcher member of RECSEM (an academic institution specialized in the study of "information systems"), has to write a paper about "*semantic information systems*" for a prestigious scientific journal. John decides to search in RECSEM's documents repositories for recent resources about this topic to build a solid knowledge background for his work. He proceeds to log into RECSEM's site and reaches the search interface page. Then, he writes a pair of keywords ("*information systems*" and "*ontologies*") in the search box, and explicitly specifies that these keywords must be searched in the abstract of each document. After that, to scope the search, John selects *"knowledge representation"* as preferred search context, *"research"* as search aim and *"3months"* as preferred date of publication, but doesn't specify any value for the *"Document type"* preference. The system checks his profile and works out that the most satisfying value for the *"Document type"* preference is *"Scientific Article"*. With this information the different agents retrieve and filter those documents that better fit to John's requirements.

The resulting set of ranked documents is displayed on John's laptop screen and looks like this:

Each document appears with an associated relevance degree and a recommendation value. This extra information eases John's task of deciding which resources can be more useful for him. In this case he decides to choose the first

Table 1. Query answer sample

D^h_j	R^h_j	T^h_j
http://www.ugr.es/~Arep/N0021	0.95	high
http://www.ugr.es/~Arep/L57641	0.93	NULL
http://www.ugr.es/~Erep/P70435	0.87	medium

document of the list (D_1) because its relevance degree and recommendation are high. After casting an evaluative glance over the resource, and before John can check another document, the system asks him if he wants to appraise the resource (thus helping other users in future search processes). John decides to cooperate and assess this document selecting the linguistic label $t' =$ *"very high"*. This appraisal, together with his user ID and the *"Search context"* preference value (*"knowledge representation"*) are added as a new "historical" recommendation item in the RF associated to D_1, ready to be used in future search processes.

Before John can leave the system, the interface agent asks him to express his opinion with respect to the global answer provided to his query. John, evaluating the quality of the resources he has checked out, considers that, in general, the results have been highly satisfactory, so he decides to select the linguistic label e_j = *"high"* (where $e_j \in S' = \{$*Total, veryHigh, high, medium, low, veryLow, Null*$\}$). Then, the system proceeds to update John's profile rewarding the frequencies of the preference values used in this query. For example, let's focus on the *"Search context"* preference. The expression domain for this preference is $F_2 = \{$*decision support systems, information retrieval systems, geographical information systems, data mining, knowledge representation, all*$\}$, and its associated frequency in John's profile is $f^j_{52} =$ *"aFewTimes"* (where $f^j_{li} \in S = \{$*always, almostAlways, mostTimes, sometimes, aFewTimes, almostNever, never*$\}$) . Taking into account that $s_a \leqslant s_b$, the values of the indexes a and b are 2 and 4 respectively, and $T = 6$, we have that $\beta = 1$. So, the frequency for this preference value is increased in a factor of one, and therefore $(f^j_{52})' = g$ (*high*, *aFewTimes*) = "*sometimes*".

After the search process, John finishes his session with the feeling that he has saved a lot of time, and with the certainty that he has obtained relevant and useful resources for his purposes.

5 Concluding Remarks

In this paper, we have described the architecture and elements of a fuzzy linguistic multi-agent system specially designed to perform information retrieval and filtering tasks in domain dependant environments.

The key aspect of the system is the joint application of Semantic Web technologies and the use of user profiles. Semantic Web technologies provide the system with the necessary semantic infrastructure to improve inference and communication capacities of agents and with means to represent the information (resources and user profiles) in a common vocabulary both human and machine interpretable. The use of user profiles allows to characterize better the user and in such a way the performance of the system can be increased. Furthermore, we have proposed a method to dynamically update user profiles that allows adapting them to the changes observed in users' preferences by mean of fuzzy linguistic matching functions.

Our plans for future work include the development of an enhanced user profile updating process based on web usage analysis and rule discovery techniques, and the adaptation of this system to different work contexts such as e-commerce or e-learning. The application of stereotypes to classify users' profiles will also be contemplated.

References

1. C. Basu, H. Hirsh, W. Cohen, Recommendation as classification: Using social and content-based information in recommendation. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence* (1998), 714–720
2. D. Beckett, RDF/XML Syntax Specification (Revised). W3C Recommendation. (10 February 2004). Available at *http://www.w3.org/TR/rdf-syntax-grammar/*. Accessed 02/19/2005
3. T. Berners-Lee, Semantic Web Road map (1998). Available at *http://www.w3.org/DesignIssues/Semantic.html*. Accessed 02/15/2005
4. T. Berners-Lee, Semantic Web - XML2000 (2000). Available at *http://www.w3.org/2000/Talks/1206-xml2k-tbl/*. Accessed 02/13/2005
5. T. Berners-Lee, J. Hendler, O. Lassila, The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities, *Scientific American* May (2001)
6. D. Bolchini, E.S.K. Yu, Modeling user requirements for Web application design, *Studies in Communication Sciences* 4 (2) (2004), 173–196
7. W. Brenner, R. Zarnekow, H. Witting, Intelligent Software Agent, Foundations and Applications, (Springer-Verlag. Berlin Heidelberg, 1998)
8. Y.A. Bishr, H. Pundt, W. Kuhn, M. Radwan, Probing the concept of information communities: a first step toward semantic interoperability. In M. Goodchild, M. Egen-Hofer, R. Fegeas, C. Kottman (eds), *Interoperating Geographic Information Systems*, (Kluwer Academic, 1999), 55–69
9. M. Claypool, A. Gokhale, T. Miranda, Combining content-based and collaborative filters in an online newspaper. In *Proceedings of the ACM SIGIR Workshop on Recommender Systems-Implementation and Evaluation* (1999)
10. 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 Word*, (John Wiley & Son, 2001), 251–276
11. 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 Computing* 6 (2002), 320–328

12. Esperonto Project. Available at *http://esperonto.semanticweb.org*. Accessed 02/21/2005
13. B. Fazlollahi, R.M. Vahidov, R.A. Aliev, Multi-agent distributed intelligent system based on fuzzy decision making, *International Journal of Intelligent Systems* 15 (2000), 849–858
14. J. Ferber, Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence (Addison-Wesley Longman, New York, 1999)
15. 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. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence* (1999), 439–446.
16. T.R. Gruber, Toward principles for the design of ontologies used for knowledge sharing. In *International Journal of Human-Computer Studies* 43 (5-6) (1995), 907–928
17. N. Guarino, Formal ontology and information systems. In N. Guarino (ed) *Formal Ontology in Information Systems, Proceedings of FOIS'98*. Trento (Italy), (IOS Press, Amsterdam, 1998), 3–17.
18. R. Guha, R. McCool, E. Miller, Semantic search. 12th International World Wide Web Conference 2003 (WWW2003), Budapest (Hungary), (2003) 700–709
19. J. Hendler, Is there an intelligent agent in your future? (1999). Available at http://www.nature.com/nature/webmatters/agents/agents.html. Accessed 02/20/2005
20. J. Hendler, Agents and the Semantic Web, *IEEE Intelligent Systems*, March, April (2001), 30–37
21. F. Herrera, E. Herrera-Viedma, J.L. Verdegay, A model of Consensus in Group Decision Problems under Linguistic Assessments, *Fuzzy Sets and Systems* 78 (1996), 73–87
22. 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.
23. E. Herrera-Viedma, Modelling the Retrieval Process of an Information Retrieval System Using an Ordinal Fuzzy Linguistic Approach, *Journal of the American Society for Information Science and Technology (JASIST)* 52 (6) (2001), 460–475
24. E. Herrera-Viedma, E. Peis, Evaluating the informative quality of documents in SGML format using fuzzy linguistic techniques based on computing with words, *Information Processing and Management* 39 (2) (2003), 195–213
25. E. Herrera-Viedma, F. Herrera, L. Martínez, J.C. Herrera, A.G. López, Incorporating Filtering Techniques in a Fuzzy Multi-Agent Model for Gathering of Information on the Web, *Fuzzy Sets and Systems* 148 (1) (2004), 61–83
26. N. Jennings, K. Sycara, M. Wooldridge, A roadmap of agent research and development, *Autonomous Agents and Multi-Agents Systems* 1 (1998), 7–38
27. P. Maes, Agents that reduce the work and information overload, *Communications of the ACM*, 37 (7) (1994), 30–40
28. D.L. McGuinness, F. van Harmelen, OWL Web Ontology Language Overview. W3C Recommendation. 10 February 2004, (2004). Available at http://www.w3.org/TR/2004/REC-owl-features-20040210/. Accessed 02/16/2005
29. 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-Verlag, 1999), 293–322

30. Ontoknowledge Project. Available at http://www.ontoknowledge.org/. Accessed 02/18/2005
31. A. Popescul, L.H. Ungar, D.M. Pennock, S. Lawrence, Probabilistic models for unified-collaborative and content-based recommendation in sparse-data environments, *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence (UAI)*, San Francisco (2001), 437–444
32. P. Reisnick, H.R. Varian, Recommender Systems. *Special issue of Comm. of the ACM* 40 (3) (1997)
33. J.B. Schafer, J.A. Konstan, J. Riedl, E-Commerce recommendation applications. *DataMining and Knowledge Discovery* 5 (1/2) (2001), 115–153
34. Semantic Web Advanced Development for Europe (SWAD-Europe). Available at http://www.w3.org/2001/sw/Europe/. Accessed 02/17/2005
35. SEWASIE Home. Available at http://www.sewasie.org. Accessed 02/17/2005
36. U. Shah, T. Finin, Y. Peng, J. Mayfield, Information Retrieval on the Semantic Web, *Proceedings of the 10*th *International Conference on Information and Knowledge Management* (2002), 461–468
37. B. Shapira, U. Hanani, A. Raveh, P. Shoval, Information filtering: A new two-phase model using stereotypic user profiling, *Journal of Intelligent Information Systems* 8 (1997), 155–165
38. Wonder Web. Available at http://wonderweb.semanticweb.org. Accessed 02/22/2005
39. R.R. Yager, Intelligent agents for World Wide Web advertising decisions, *International Journal of Intelligent Systems* 12 (1997), 379–390
40. L.A. Zadeh, The concept of a linguistic variable and its applications to approximate reasoning. Part I, In *Information Sciences* 8 (1975), 199–249. Part II, In *Information Sciences* 8 (1975), 301–357. Part III, In *Information Sciences* 9 (1975), 43–80
41. Y. Zou, T. Finin, L. Ding, H. Chen, R. Pan, Using semantic web technology in multi-agent systems: a case study in the TAGA trading agent environment. In *ACM International Conference Proceeding Series. Proceedings of the 5th international conference on Electronic commerce* (2003), 95–101