A Semantic Model of Selective Dissemination of Information for Digital Libraries

J. M. Morales-del-Castillo, R. Pedraza-Jiménez, A. A. Ruíz, E. Peis, and E. Herrera-Viedma

In this paper we present the theoretical and methodological foundations for the development of a multi-agent Selective Dissemination of Information (SDI) service model that applies Semantic Web technologies for specialized digital libraries. These technologies make possible achieving more efficient information management, improving agent–user communication processes, and facilitating accurate access to relevant resources. Other tools used are fuzzy linguistic modelling techniques (which make possible easing the interaction between users and system) and natural language processing (NLP) techniques for semiautomatic thesaurus generation. Also, RSS feeds are used as "current awareness bulletins" to generate personalized bibliographic alerts.

Nowadays, one of the main challenges faced by information systems at libraries or on the Web is to efficiently manage the large number of documents they hold. Information systems make it easier to give users access to relevant resources that satisfy their information needs, but a problem emerges when the user has a high degree of specialization and requires very specific resources, as in the case of researchers.[1] In "traditional" physical libraries, several procedures have been proposed to try to mitigate this issue, including the selective dissemination of information (SDI) service model that make it possible to offer users potentially interesting documents by accessing users' personal profiles kept by the library.

Nevertheless, the progressive incorporation of new information and communication technologies (ICTs) to information services, the widespread use of the Internet, and the diversification of resources that can be accessed through the Web has led libraries through a process of reinvention and transformation to become "digital" libraries.[2] This reengineering process requires a deep revision of work techniques and methods so librarians can adapt to the new work environment and improve the services provided.

In this paper we present a recommendation and SDI model, implemented as a service of a specialized digital library (in this case, specialized in library and information science), that can increase the accuracy of accessing information and the satisfaction of users' information needs on the Web.

This model is built on a multi-agent framework, similar to the one proposed by Herrera-Viedma, Peis, and Morales-del-Castillo,[3] that applies Semantic Web technologies within the specific domain of specialized digital libraries in order to achieve more efficient information management (by semantically enriching different elements of the system) and improved agent–agent and user–agent communication processes.

Furthermore, the model uses fuzzy linguistic modelling techniques to facilitate the user–system interaction and to allow a higher grade of automation in certain procedures. To increase improved automation, some natural language processing (NLP) techniques are used to create a system thesaurus and other auxiliary tools for the definition of formal representations of information resources.

In the next section, "Instrumental basis," we briefly analyze SDI services and several techniques involved in the Semantic Web project, and we describe the preliminary methodological and instrumental bases that we used for developing the model, such as fuzzy linguistic modelling techniques and tools for NLP. In "Semantic SDI service model for digital libraries," the bulk of this work, the application model that we propose is presented. Finally, to sum up, some conclusive data are highlighted.

Instrumental basis

Filtering techniques for SDI services

Filtering and recommendation services are based on the application of different process-management techniques that are oriented toward providing the user exactly the information that meets his or her needs or can be of his or her interest. In textual domains, these services are usually developed using multi-agent systems, whose main aims are

- to evaluate and filter resources normally represented in XML or HTML format; and
- to assist people in the process of searching for and retrieving resources.[4]

J. M. Morales-del-Castillo (josemdc@ugr.es) is Assistant Professor of Information Science, Library and Information Science Department, University of Granada, Spain. **R. Pedraza-Jiménez** (rafael.pedraza@upf.edu) is Assistant Professor of Information Science, Journalism and Audiovisual Communication Department, Pompeu Fabra University, Barcelona, Spain. **A. A. Ruíz** (aangel@ugr.es) is Full Professor of Information Science, Library and Information Science Department, University of Granada. **E. Peis** (epeis@ugr.es) is Full Professor of Information Science, Library and Information Science Department, University of Granada. **E. Herrera-Viedma** (viedma@decsai.ugr.es) is Senior Lecturer in Computer Science, Computer Science and Artificial Intelligence Department, University of Granada.

Traditionally, these systems are classified as either content-based recommendation systems or collaborative recommendation systems.[5] Content-based recommendation systems filter information and generate recommendations by comparing a set of keywords defined by the user with the terms used to represent the content of documents, ignoring any information given by other users. By contrast, collaborative filtering systems use the information provided by several users to recommend documents to a given user, ignoring the representation of a document's content. It is common to group users into different categories or stereotypes that are characterized by a series of rules and preferences, defined by default, that represent the information needs and common behavioural habits of a group of related users. The current trend is to develop hybrids that make the most of content-based and collaborative recommendation systems.

In the field of libraries, these services usually adopt the form of SDI services that, depending on the profile of subscribed users, periodically (or when required by the user) generate a series of information alerts that describe the resources in the library that fit a user's interests.[6]

SDI services have been studied in different research areas, such as the multi-agent systems development domain,[7] and, of course, the digital libraries domain.[8] Presently, many SDI services are implemented on Web platforms based on a multi-agent architecture where there is a set of intermediate agents that compare users' profiles with the documents, and there are input-output agents that deal with subscriptions to the service and display generated alerts to users.[9] Usually, the information is structured according to a certain data model, and users' profiles are defined using a series of keywords that are compared to descriptors or the full text of the documents.

Despite their usefulness, these services have some deficiencies:

- The communication processes between agents, and between agents and users, are hindered by the different ways in which information is represented.
- This heterogeneity in the representation of information makes it impossible to reuse such information in other processes or applications.

A possible solution to these deficiencies consists of enriching the information representation using a common vocabulary and data model that are understandable by humans as well as by software agents. The Semantic Web project takes this idea and provides the means to develop a universal platform for the exchange of information.[10]

Semantic Web technologies

The Semantic Web project tries to extend the model of the present Web by using a series of standard languages that enable enriching the description of Web resources and make them semantically accessible.[11] To do that, the project basis itself on two fundamental ideas: (1) resources should be tagged semantically so that information can be understood both by humans and computers, and (2) intelligent agents should be developed that are capable of operating at a semantic level with those resources and that infer new knowledge from them (shifting from the search of keywords in a text to the retrieval of concepts).[12]

The semantic backbone of the project is the Resource Description Framework (RDF) vocabulary, which provides a data model to represent, exchange, link, add, and reuse structured metadata of distributed information sources, thereby making them directly understandable by software agents.[13] RDF structures the information into individual assertions (e.g., "resource," "property," and "property value triples") and uniquely characterizes resources by means of Uniform Resource Identifiers (URIs), allowing agents to make inferences about them using Web ontologies or other, simpler semantic structures, such as conceptual schemes or thesauri.[14]

Even though the adoption of the Semantic Web and its application to systems like digital libraries is not free from trouble (because of the nature of the technologies involved in the project and because of the project's ambitious objectives,[15] among other reasons), the way these technologies represent the information is a significant improvement over the quality of the resources retrieved by search engines, and it also allows the preservation of platform independence, thus favouring the exchange and reuse of contents.[16]

As we can see, the Semantic Web works with information written in natural language that is structured in a way that can be interpreted by machines. For this reason, it is usually difficult to deal with problems that require operating with linguistic information that has a certain degree of uncertainty (e.g., when quantifying the user's satisfaction in relation to a product or service). A possible solution could be the use of fuzzy linguistic modelling techniques as a tool for improving system–user communication.

Fuzzy linguistic modelling

Fuzzy linguistic modelling supplies a set of approximate techniques appropriate for dealing with qualitative aspects of problems.[17] The ordinal linguistic approach is defined according to a finite set of tags *(S)* completely ordered and with odd cardinality (seven or nine tags):

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

The central term has a value of approximately 0.5, and the rest of the terms are arranged symmetrically around

it. The semantics of each linguistic term is given by the ordered structure of the set of terms, considering that each linguistic term of the pair (s_i, s_{T-i}) is equally informative. Each label s_i is assigned a fuzzy value defined in the interval [0,1] that is described by a linear trapezoidal property function represented by the 4-tupla $(a_i, b_i, \alpha_i, \beta_i)$. (The two first parameters show the interval where the property value is 1.0; the third and fourth parameters show the left and right limits of the distribution.) Additionally, we need to define the following properties:

1.–The set is ordered: $s_i \geq s_j$ *if* $i \geq j$.

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* $s_i \geq s_j$.

4.–Minimization operator: $MIN(s_i, s_j) = s_i$ *if* $s_i \leq s_j$.

It also is necessary to define aggregation operators, such as Linguistic Weighted Averaging (LWA),[18] capable of and operating with and combining linguistic information.

Focusing on facilitating the interaction between users and system, the other starting objective is to achieve the development and implementation of the model proposed in the most automated way possible. To do this, we use a basic auxiliary tool—a thesaurus—that, among other tasks, assists users in the creation of their profile and enables automating the alerts generation.

That is why it is critical to define the way in which we create this tool, and in this work we propose a specific method for the semiautomatic development of thesauri using NLP techniques.

NLP techniques and other automating tools

NLP consists of a series of linguistic techniques, statistic approaches, and machine learning algorithms (mainly clustering techniques) that can be used, for example, to summarize texts in an automatic way, to develop automatic translators, and to create voice recognition software.

Another possible application of NLP would be the semiautomatic construction of thesauri using different techniques. One of them consists of determining the lexical relations between the terms of a text (mainly synonymy, hyponymy, and hyperonymy),[19] and extracting terms that are more representative for the text's specific domain.[20] It is possible to elicit these relations by using linguistic tools, like Princeton's WordNet (http://wordnet.princeton.edu) and clustering techniques.

WordNet is a powerful multilanguage lexical database where each one of its entries is defined, among other elements, by their synonyms (synsets), hyponyms, and hyperonyms.[21] As a consequence, once given the most important terms of a domain, WordNet can be used to create from them a thesaurus (after leaving out all terms that have not been identified as belonging or related to the domain of interest).[22]

This tool can also be used with clustering techniques—for example, to group documents of a collection in a set of nodes or clusters, depending on their similarity. Each of these clusters is described by the most representative terms of their documents. These terms make up the most specific level of a thesaurus and are used to search in WordNet for their synonyms and most general terms, contributing (with the repetition of this procedure) to the bottom-up-development process of the thesaurus.[23]

Although there are many others, these are some of the most well-known techniques of semiautomatic thesaurus generation (semiautomatic because, needless to say, the supervision of experts is necessary to determine the validity of the final result).

For specialized digital libraries, we propose developing, on a multi-agent platform and using all these tools, SDI services capable of generating alerts and recommendations for users according to their personal profiles. In particular, the model presented here is the result of several previous models merging, and its service is based on the definition of "current-awareness bulletins," where users can find a basic description of the resources recently acquired by the library or those that might be of interest to them.[24]

The Semantic SDI service model for digital libraries

The SDI service includes two agents (an interface agent and a task agent) distributed in a four-level hierarchical architecture: user level, interface level, task level and resource level.

Its main components are a repository of full-text documents (which make up the stock of the digital library) and a series of elements described using different RDF-based vocabularies: one or several RSS feeds that play a role similar to that of current-awareness bulletins in traditional libraries; a repository of recommendation log files that store the recommendations made by users about the resources, and a thesaurus that lists and hierarchically relates the most relevant terms of the specialization domain of the library.[25] Also, the semantics of each element (that is, its characteristics and the relations the element establishes with other elements in the system) are defined in a Web ontology developed in Web Ontology Language (OWL).[26]

Next, we describe these main elements as well as the different functional modules that the system uses to carry out its activity.

Elements of the model

There are four basic elements that make up the system:

the thesaurus, user profiles, RSS feeds, and recommendation log files.

Thesaurus

An essential element of this SDI service is the thesaurus, an extensible tool used in traditional libraries that enables organizing the most relevant concepts in a specific domain, defining the semantic relations established between them, such as equivalence, hierarchical, and associative relations. The functions defined for the thesaurus in our system include helping in the indexing of RSS feeds items and in the generation of information alerts and recommendations.

To create the thesaurus, we followed the method suggested by Pedraza-Jiménez, Valverde-Albacete, and Navia-Vázquez.[27]

The learning technique used for the creation of a thesaurus includes four phases: preprocessing of documents, parameterizing the selected terms, conceptualizing their lexical stems, and generating a lattice or graph that shows the relation between the identified concepts.

Essentially, the aim of the preprocessing phase is to prepare the documents' parameterization by removing elements regarded as superfluous. We have developed this phase in three stages: eliminating tags (stripping), standardizing, and stemming.

In the first stage, all the tags (HTML, XML, etc.) that can appear in the collection of documents are eliminated. The second stage is the standardization of the words in the documents in order to facilitate and improve the parameterization process. At this stage, the acronyms and N-grams (bigrams and trigrams) that appear in the documents are identified using lists that were created for that purpose.

Once we have detected the acronyms and N-grams, the rest of the text is standardized. Dates and numerical quantities are standardized, being substituted with a script that identifies them. All the terms (except acronyms) are changed to small letters, and punctuation marks are removed. Finally, a list of function words is used to eliminate from the texts articles, determiners, auxiliary verbs, conjunctions, prepositions, pronouns, interjections, contractions, and grade adverbs.

All the terms are stemmed to facilitate the search of the final terms and to improve their calculation during parameterization. To carry out this task, we have used Morphy, the stemming algorithm used by WordNet. This algorithm implements a group of functions that check whether a term is an exception that does not need to be stemmed and then convert words that are not exceptions to their basic lexical form. Those terms that appear in the documents but are not identified by Morphy are eliminated from our experiment.

The parameterization phase has a minimum complexity. Once identified, the final terms (roots or bases) are quantified by being assigned a weight. Such weight is obtained by the application of the scheme *term frequency-inverse document frequency (tf-idf)*, a statistic measure that makes possible the quantification of the importance of a term or N-gram in a document depending on its frequency of appearance and in the collection the document belongs to.

Finally, once the documents have been parameterized, the associated meanings of each term (lemma) are extracted by searching for them in WordNet (specifically, we use WordNet 2.1 for UNIX-like systems). Thus we get the group of synsets associated with each word. The group of hyperonyms and hyponyms also are extracted from the vocabulary of the analyzed collection of documents.

The generation of our thesaurus—that is, the identification of descriptors that better represent the content of documents, and the identification of the underlying relations between them—is achieved using formal concept analysis techniques.

This categorization technique uses the theory of lattices and ordered sets to find abstraction relations from the groups it generates. Furthermore, this technique enables clustering the documents depending on the terms (and synonyms) it contains. Also, a lattice graph is generated according to the underlying relations between the terms of the collection, taking into account the hyperonyms and hyponyms extracted. In that graph, each node represents a descriptor (namely, a group of synonym terms) and clusters the set of documents that contain it, linking them to those with which it has any relation (of hyponymy or hyperonymy).

Once the thesaurus is obtained by identifying its terms and the underlying relations between them, it is automatically represented using the Simple Knowledge Organization System (SKOS) vocabulary (see figure 1).[28]

User profiles

User profiles can be defined as structured representations that contain personal data, interests, and preferences of users with which agents can operate to customize the SDI service. In the model proposed here, these profiles are basically defined with Friend of a Friend (FOAF), a specific RDF/XML for describing people (which favours the profile interoperability, since this is a widespread vocabulary supported by an OWL ontology) and another nonstandard vocabulary of our own to define fields not included in FOAF (see figure 2).[29]

Profiles are generated the moment the user is registered in the system, and they are structured in two parts: a public profile that includes data related to the user's identity and affiliation, and a private profile that includes the user's interests and preferences about the topic of the alerts he or she wishes to receive.

To define their preferences, users must specify keywords and concepts that best define their information

```
<skos:Concept rdf:about="7">
  <skos:inScheme rdf:resource="http://www.ugr.es/…/thes/"/>
  <skos:prefLabel xml:lang="es">Proceedings</skos:prefLabel>
  <skos:broader rdf:resource="http://www.ugr.es/…/thes/668"/>
  <skos:narrower rdf:resource="http://www.ugr.es/…/thes/286"/>
  <skos:narrower rdf:resource="http://www.ugr.es/…/thes/830"/>
</skos:Concept>
```

Figure 1. Sample entry of a SKOS Core thesaurus

needs. Later, the system compares those concepts with the terms in the thesaurus using as a similarity measure the edit tree algorithm.[30] This function matches character strings, then returns the term introduced (if there's an exact match) or the lexically most similar term (if not).

Consequently, if the suggested term satisfies user expectations, it will be added to the user's profile together with its synonyms (if any). In those cases where the suggested term is not satisfactory, the system must have any tool or application that enables users to browse the thesaurus and select terms that better describe their needs. An example of this type of applications is ThManager (http://thmanager.sourceforge.net), a project of the Universidad de Zaragoza, Spain, that enables editing, visualizing, and going through structures defined in SKOS.

Each of the terms selected by the user to define his or her areas of interest has an associated linguistic frequency value (tagged as <freq>) that we call "satisfaction frequency." It represents the regularity with which a particular preference value has been used in alerts positively evaluated by the user. This frequency measures the relative importance of the preferences stated by the user and allows the interface agent to generate a ranking list of results. The range of possible values for these frequencies is defined by a group of seven labels that we get from the fuzzy linguistic variable "Frequency," whose expression domain is defined by the linguistic term set S = *{always, almost_always, often, occasionally, rarely, almost_never, never}*, being the default value and "occasionally" being the central value.

RSS feeds

Thanks to the popularization of blogs, there has been widespread use of several vocabularies specifically designed for the syndication of contents (that is, for making accessible to other Internet users the content of a website by means of hyperlink lists called "feeds"). To create our current-awareness bulletin we use RSS 1.0, a vocabulary that enables managing hyperlinks lists in an easy and flexible way. It utilizes the RDF/XML syntax and data model and is easily extensible because of the use of

```
<foaf:PersonalProfileDocument rdf:about="">
  <foaf:maker rdf:resource="#person"/>
  <foaf:primaryTopic rdf:resource="#person"/>
</foaf:PersonalProfileDocument>
<foaf:Person rdf:ID="user_09234">
  <foaf:name>Diego Allione</foaf:name>
  <foaf:title>Sr.</foaf:title>
  <foaf:mbox_sha1sum>af9fa7601df46e95566</foaf:mbox_sha1sum>
  <foaf:homepage rdf:resource="http://allione.org"/>
  <foaf:depiction rdf:resource="allione.jpg"/>
  <foaf:phone rdf:resource="tel:555-432-432"/>
  <dfss:topic>
    <dfss:pref rdf:nodeID="pref_09234-1">
      <rdfs:label>Library management</rdfs:label>
      <dfss:relev>0.83</dfss:relev>
    </dfss:pref>
  </dfss:topic>
</foaf:Person>
```

Figure 2. User profile sample

modules that enable extending the vocabulary without modifying its core each time new describing elements are added. In this model several modules are used: the Dublin Core (DC) module to define the basic bibliographic information of the items utilizing the elements established by the Dublin Core Metadata Initiative (http://dublincore.org); the syndication module to facilitate software agents synchronizing and updating RSS feeds; and the taxonomy module to assign topics to feeds items.

The structure of the feeds comprises two areas: one where the channel itself is described by a series of basic metadata like a title, a brief description of the content, and the updating frequency; and another where the descriptions of the items that make up the feed (see figure 3) are defined (including elements such as title, author, summary, hyperlink to the primary resource, date of creation, and subjects).

```
<item rdf:about="http://www.ugr.es/…/doc-00000528">
    <dc:creator>Escudero Sánchez, Manuel</dc:creator>
    <dc:creator>Fernández Cáceres, José Luis</dc:creator>
    <title>Broadcasting and the Internet</title>
    <link>http://eprints.rclis.org/…/AudioVideo_good.pdf</link>
    <description>This paper is about…</description>
    <dc:date>2002</dc:date>
    <dc:source>REDOC, 8 (4), 2008</dc:source>
    <dc:subject xml:lang="en">Virual communities</dc:subject>
</item>
```

Figure 3. RSS feed item sample

Recommendation log file

Each document in the repository has an associated recommendation log file in RDF that includes the listing of evaluations assigned to that resource by different users since the resource was added to the system. Each of the entries of the recommendation log files consists of a recommendation value, a URI that identifies the user that has done the recommendation, and the date of the record (see figure 4). The expression domain of the recommendations is defined by the following set of five fuzzy linguistic labels that are extracted from the linguistic variable "Quality of the resource": Q = {*Very_low, Low, Medium, High, Very_high*}.

```
<recomm-log rdf:ID="log-00528">
    <doc rdf:resource="http://doc.es/doc-0A15"/>
    <items_e>
      <item rdf:nodeID="item-000A901">
        <user rdf:resource="http://user.es/001"/>
        <date>14/03/2007</date>
        <recomm>High</recomm>
      </item>
    </ítems_e>
</recomm-log>
```

Figure 4. Recommendation log file sample

These elements represent the raw materials for the SDI service that enable it to develop its activity through four processes or functional modules: the profiles updating process, RSS feeds generation process, alert generation process, and collaborative recommendation process.

System processes

Profiles updating process

Since the SDI service's functions are based on generating passive searches to RSS feeds from the preferences stored in a user's profile, updating the profiles becomes a critical task. User profiles are meant to store long-term preferences, but the system must be able to detect any subtle change in these preferences over time to offer accurate recommendations.

In our model, user profiles are updated using a simple mechanism that enables finding users' implicit preferences by applying fuzzy linguistic techniques and taking into account the feedback users provide. Users are asked about their satisfaction degree (e_j) in relation to the information alert generated by the system (i.e., whether the items

retrieved are interesting or not). This satisfaction degree is obtained from the linguistic variable "Satisfaction," whose expression domain is the set of five linguistic labels: S′ = {*Total, Very_high, High, Medium, Low, Very_low, Null*}.

This mechanism updates the satisfaction frequency associated with each user preference according to the satisfaction degree e_j. It requires the use of a matching function similar to those used to model threshold weights in weighted search queries.[31] The function proposed here rewards the frequencies associated with the preference values present when resources assessed are satisfactory, and it penalizes them when this assessment is negative. Let $e_j \in S'$ be the degree of satisfaction, and $f^j_{il} \in S$ the frequency of property i (in this case i = "Preference") with value l, then we define the updating function g as S′x S→S:

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

$$s_a, s_b \in S \mid a, b \in H = \{0,...,T\}$$

where, (i) $s_{a=} f^{j}{}_{li}$; (ii) $s_{b=} e_j$; (iii) a and b are indexes of linguistic labels whose value ranges from 0 to T (being T the cardinality of the group S minus one), and β (iv) is a bonus value defined as ß = *round* $(2 \mid b\text{-}a \mid / T)$ that rewards or penalizes the satisfaction frequencies.

The more resources the user asseses, the more precise the mechanism becomes, since it will be easier for the SDI service to "learn" which documents are likely more interesting to the user according to his or her preferences. This evaluation process is not only useful for updating users' profiles: As we will see in the next section, the feedback provided by the users can be reused to define a recommendation system that benefits from the their experience, knowledge, and critical skills.

RSS feeds generation and updating process

In this module, the RSS feeds of the digital library (one or several feeds, depending on the library's specific needs) are created and updated semiautomatically (see figure 5). Therefore the system administrator must play an active role in the process, defining through a simple input interface the different elements needed to describe each RSS feed and their corresponding items. This task can be simplified if the administrator is able to complete the description of both feeds and items that use any application capable of extracting metadata from resources in databases or online repositories. In this specific case, we have chosen to use DigiDocMeta, a tool designed by the Pompeu Fabra University's DigiDoc Laboratory (www.metaeditor.net) that analyzes the content of resources and subsequently supplies the administrator with descriptive data (such as title, summary, keywords, and language) extracted automatically from them. Administrators can check those data through an easy and clear interface, and, if necessary, they can use different edition tools to modify them.

However, in both cases, the system administrator must supervise the assignment of topics describing the content of the resource. To facilitate this task, we also use a tool that helps in the process of assigning topics to items. It works in a similar way to the preferences selection process (as described earlier): The administrator suggests a series of topics that are lexically matched to the terms of the thesaurus using the tree edit algorithm. The terms that exactly match the terms of the thesaurus will be assigned as a topic (together with their synonyms). If there's no match, the system will suggest a series of lexically similar terms that the administrator can use or not, depending on his or her own opinion.

Alert generation or information push process

This module can be considered the backbone of the SDI service (see figure 6). It consists of triggering against the RSS feed a passive search query (i.e., on behalf of the user) about the areas of interest defined in his or her profile. Consequently, the customized information alert generated by the system will be displayed without an explicit request from the user (this is known as "push," or passive reception of information). This process consists of four steps:

1. Users access the system by giving their username and password.
2. The task agent compares the areas of interest in the active user's profile with the descriptors that characterize the content of the n items of the RSS feed. The agent distinguishes terms that better meet the specific information needs of the user. In this case, instead of using traditional lexical matching (where two character strings are compared), we propose using a semantic similarity measure and harnessing the thesaurus as a tool for organizing knowledge. To do so, we use a function, defined by Oldakowsky and Byzer, that enables calculating the similarity between RDF objects.[32] This function makes it possible to determine the distance between two terms in a concept schema according to their situation within the conceptual hierarchy (in our model, this concept schema is the thesaurus of the system).
3. The task agent presents to the user those resources whose similarity matches or exceeds a predefined threshold, k (k being a value close to 1), discarding any document that does not reach it.
4. The interface agent generates an alert in the homepage of the website that notifies the user of the existence of new documents that can be of interest to him or her. This alert links directly to the list generated by the task agent, from which the

user can access to the different resources in full text. If there is more than one RSS feed in the library, the retrieved items from each feed must be aggregated into a single list of results. If no relevant items are found, the user will be notified.

Collaborative recommendation process

This system can also offer additional information about the recommended resources, regardless of the similarity between content descriptors and users' preferences. This can be achieved by defining an auxiliary collaborative recommendation system that is based on the opinions of users in the library with a profile similar to that of the active user.

Therefore, taking as a starting point the list of recommended resources, the task agent retrieves the associated recommendation log file of each item and extracts both the identifier of all the users that have ever recommended that specific resource, and the corresponding assessment given by each one of them (we have to keep in mind that these assessments are based on the satisfaction degree stated by the user in the profile updating process).

The next step is comparing the profile of each user with the active user's profile in a way similar to the process of information push (but matching preferences instead of topics and preferences this time). In other words, the task agent proceeds to define a cluster of similar users.

Finally, the task agent aggregates the different assessments using the fuzzy linguistic operator LWA,[33] which returns as output a new linguistic tag that gives users new criteria to select resources of interest.

Conclusions and future works

Libraries are moving services (like SDI) to the Web. Combining Semantic Web technologies with NLP techniques and fuzzy linguistic techniques favours the development of improved SDI services that are capable of offering accurate information according to users' needs.

The Semantic Web has a common data model and syntax that guarantee the interoperability of resources (independently of the platform), thus making easier the establishment of exchange and collaborative networks between digital libraries. Furthermore, these technologies make it possible to considerably improve the communication processes between agents and between users and agents.

Because NLP techniques and formal concept analysis enable detecting descriptors and, in combination with other lexical resources, identifying the semantic relations among them (synonymy, hyponymy and hyperonymy relations in particular), they facilitate the semiautomatic generation of thesauri. These thesauri can later be used as tools for the semiautomatic indexation of resources and to generate alerts and recommendations. However, those tasks require of the attentive supervision of a system administrator, who is responsible for deciding whether the suggested subjects or keywords are appropriate.

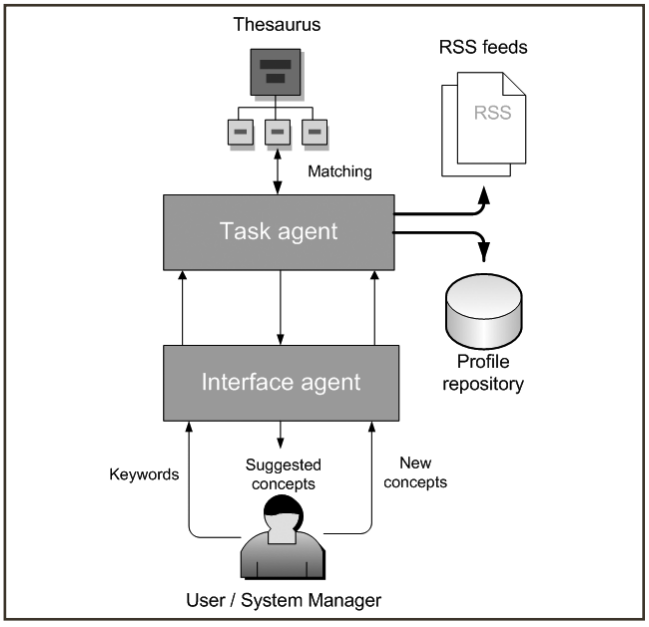

Figure 5. Profiles and RSS feeds generation process

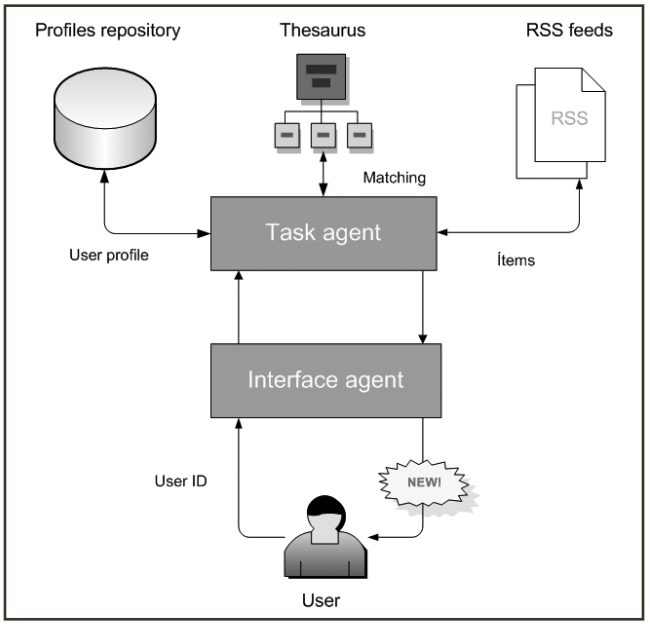

Figure 6. Alert generation process

On the other hand, the use of auxiliary tools such as metadata extractors makes possible the formalization of resource descriptions that will be later spread through RSS feeds, as well as the modification of the suggested descriptions and the inclusion of additional information to those descriptions.

A future line of research should be focused on the development of an integrated application that makes possible the semiautomatic generation of thesauri by following the methods described in this work. This way, each information center or library would have the capability of generating its own specialized and domain-dependant thesaurus. It would be done starting from the full-text electronic documents available in the library, as well as any other collection of documents or specialized digital information sources, such as websites, specialized dictionaries, and so on.

Acknowledgement

This work has been supported by the Research Projects TIN2007-61079 and SAINFOWEB-PAI00602.

References

1. K. D. Bollacker, S. Lawrence, and C. L. Giles, "Discovering Relevant Scientific Literature on the Web," *IEEE Intelligent Systems* 15, no. 2 (2000): 42–47.

2. G. Marchionini, "Research and Development in Digital Libraries," http://ils.unc.edu/~march/digital_library_R_and_D.html (accessed Nov. 23, 2008); A. F. Smeaton and J. Callan, "Personalisation and Recommender Systems in Digital Libraries," *International Journal of Digital Libraries* 5, no. 4 (2005): 299–308.

3. E. Herrera-Viedma, E. Peis, and J. M. Morales-del-Castillo, "A Fuzzy Linguistic Multi-Agent Model Based on Semantic Web Technologies and User Profiles," *Soft Computing in Web Information Retrieval*, ed. E. Herrera-Viedma, G. Pasi, and F. Crestani (Berlin/Heidelberg: Springer-Verlag, 2006): 105–20.

4. P. Resnick and H. R. Varian, "Recommender Systems," *Communications of the ACM* 40, no. 3 (1997): 56–58.

5. A. Popescul et al., "Probabilistic Models for Unified-Collaborative and Content-Based Recommendation in Sparse-Data Environments," *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence* (San Francisco: Morgan Kaufmann, 2001): 437–44.

6. D. Aksoy et al., "Research in Data Broadcast and Dissemination," *Proceedings of the 1st International Conference on Advanced Multimedia Content Processing. Lecture Notes in Computer Science, 1554* (Berlin/Heidelberg: Springer-Verlag, 1998): 194–207; P. W. Foltz and S. T. Dumais, "Personalized Information Delivery: An Analysis of Information Filtering Methods," *Communications of the ACM* 35, no. 12 (1992): 51–60.

7. K. Decker, K. Sycara, and M. Williamson, "Middle-Agents for the Internet," *Proceedings of the IJCAI-97* (Nagoya, Japan, 1997): 578–84; D. Kuokka and L. Harada, "Matchmaking for Information Agents," *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence* (Edinburgh, UK: Professional Book Center, 1995): 672–78.

8. D. Faensen et al., "Hermes: A Notification Service for Digital Libraries," *Proceedings of the Joint ACM/IEEE Conference on Digital Libraries (JCDL '01)* (New York: ACM, 2001): 373–80.

9. M. Altinel and M. J. Franklin, "Efficient Filtering of XML Documents for Selective Dissemination of Information," *Proceedings of the 26th International Conference on Very Large Data Bases* (San Francisco: Morgan Kaufmann, 2000), 53–64; T. W. Yan and H. Garcia-Molina, "The SIFT Information Dissemination System," *ACM Transactions on Database Systems* 24, no. 4 (1999): 529–65.

10. T. Berners-Lee, J. Hendler, and 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*, www.sciam.com/article.cfm?id=the-semantic-web (accessed Nov. 22, 2008).

11. T. Berners-Lee, "Semantic Web Road map," www.w3.org/DesignIssues/Semantic.html (accessed Nov. 16, 2008).

12. J. Hendler, "Agents and the Semantic Web," *IEEE Intelligent Systems* 16, no. 2 (Mar./Apr. 2001): 30–37.

13. D. Becket, ed., "RDF/XML Syntax Specification (Revised)," www.w3.org/TR/rdf-syntax-grammar (accessed Nov. 15, 2008).

14. T. Gruber, "Toward Principles for the Design of Ontologies Used for Knowledge Sharing," *International Journal of Human-Computer Studies* 43, no. 5–6 (1995): 907–28; N. Guarino, "Formal Ontology and Information Systems," *Proceedings of FOIS'98* (Amsterdam: IOS Pr., 1998): 3–17.

15. L. Codina and C. Rovira, "La Web semántica," *Tendencias en documentación digital*, ed. J. Tramullas (Gijón: Trea, 2006): 9–54.

16. R. Pedraza-Jiménez, L. Codina, and C. Rovira, "Web semántica y ontologías en el procesamiento de la información documental," El profesional de la información 16, no. 6 (2007): 569–78.

17. L. A. Zadeh, "The Concept of a Linguistic Variable and its Applications to Approximate Reasoning," part 1, *Information Sciences* 8, no. 3 (1975): 199–249; part 2, *Information Sciences* 8, no. 4 (1975): 301–57; part 3, *Information Sciences* 9, no. 1 (1975): 43–80.

18. F. Herrera and E. Herrera-Viedma, "Aggregation Operators for Linguistic Weighted Information," *IEEE Transactions on Systems, Man and Cybernetics* 27, no. 5 (1997): 646–56.

19. M. A. Hearst, "Automatic Acquisition of Hyponyms from Large Text Corpora," *Proceedings of the 14th Conference on Computational Linguistics* (Morristown, N.J., 1992), 539–45.

20. N. Aussenac-Gilles, B. Biébow, and N. Szulman, "Revisiting Ontology Design: A Method Based on Corpus Analysis," *Lecture Notes in Artificial Intelligence, 1937. Knowledge Engineering and Knowledge Management: Methods, Models and Tools, Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management* (Berlin/Heidelberg: Springer-Verlag, 2000): 172–88.

21. G. Miller, "WordNet: A Lexical Database for English," *Communications of the ACM* 38, no. 11 (1995): 39–41.

22. M. Missikof, R. Navigli, and P. Velardi, "Integrated Approach to Web Ontology Learning and Engineering," *IEEE Computer* 35, no. 11 (2002): 60–63.

23. L. Khan and F. Luo, "Ontology Construction for

Information Selection," *Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'02)* (Washington, D.C.: IEEE Computer Society, 2002): 122–31.

24. E. Peis et al., "Servicios semánticos de difusión selectiva de información (DSI) basados en RSS," *Proceedings of the 1st International Conference on Multidisciplinary Information Sciences and Technology* (Mérida, Spain, 2006): 197–201.

25. G. Beged-Dov et al., eds., "RDF Site Summary (RSS) 1.0," http://web.resource.org/rss/1.0/spec (accessed Nov. 26, 2008).

26. D. L. McGuinnes and F. van Harmelen, "OWL Web Ontology Language Overview," www.w3.org/TR/owl-features (accessed Nov. 26, 2008).

27. R. Pedraza-Jiménez, F. Valverde-Albacete, and A. Navia-Vázquez, "A Generalisation of Fuzzy Concept Lattices for the Analysis of Web Retrieval Tasks," *Proceedings of the 6th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems* (New York: Springer-Verlag, 2006): 132–41.

28. A. Miles and D. Brickley, "SKOS Core Guide," www.w3.org/TR/2005/WD-swbp-skos-core-guide-20051102 (accessed Jan. 20, 2009).

29. D. Brickley and L. Miller, eds., "FOAF Vocabulary Specification," www.xmlns.com/foaf/0.1 (accessed Nov. 20, 2008).

30. V. I. Levenshtein, "Binary Codes Capable of Correcting Deletions, Insertions, and Reversals," *Soviet Physics Doklady* 10, no. 8 (1966): 707–10.

31. E. Herrera-Viedma, "Modelling the Retrieval Process of an Information Retrieval System Using an Ordinal Fuzzy Linguistic Approach," *Journal of the American Society of Information System and Technology* 52, no. 6 (2001): 460–75.

32. R. Oldakowsky and C. Byzer, "SemMF: A Framework for Calculating Semantic Similarity of Objects Represented as RDF Graphs," http://sites.wiwiss.fu-berlin.de/suhl/radek/pub/SemMF_ISWC2005.pdf (accessed Nov. 22, 2008).

33. Herrera and Herrera-Viedma, "Aggregation Operators for Linguistic Weighted Information."