7

Digestion of Knowledge in a KM System to Reveal Implicit Knowledge

Jaime Moreno-Llorena and Xavier Alamán Roldán
Universidad Autónoma de Madrid
Spain

1. Introduction

The motivation of this work is the problem of information overload in the ICT-based systems. We think network knowledge management systems have the most important characteristics of systems with this problem, but they are more scalable and controllable than others, so they might be used as a research experimental model.
Our assumption is that there are several hidden aspects in the systems with information overload, which can be used to try to solve this problem. On the one hand, taking advantage of the excess energy of the active elements that are involved in the systems, such as users, services or applications and other entities related to them. One the other hand, using the properties of both the elements and the activities related to the systems affected by the problem, such as network, active entities, information and knowledge involved, or processes and interactions of these elements and activities.
In applying this assumption to proposed simplified experimental model of knowledge management systems, we try to discover ways to reduce information overload in these systems, which could be applied in broader areas such as the Web.
Our approach is based on a knowledge management system called KnowCat (KC) (Alamán & Cobos, 1999; Cobos, 2003; Cobos & Pifarré, 2008), which is a groupware system that facilitates the management of a knowledge repository by means of user community interaction through the Web. KC achieves a selection of the best documents without supervision by anyone, using information about users' activity and users' opinion about knowledge. KC knowledge repository is formed by documents and topics structured as a knowledge tree. Each KC instance is a KC Node and has a subject, a user community and a knowledge repository. Crystallization is the KC's process of knowledge selection by its quality using information about users' activity and their opinion about the knowledge items. For more information, please see the chapter about KC in this book.
In order to support the assumption, a prototype has been developed on KnowCat, which is called Semantic KnowCat (SKC) (Moreno-Llorena, 2008; Moreno-Llorena & Alamán, 2005) incorporating ideas and techniques from different research areas that converge on the Semantic Web (Berners-Lee, 2000): Knowledge Management, Human Computer Interaction, Collaborative Work, and Data and Information Mining (Baeza & Ribeiro, 1999).
This article shows how some of the techniques and ideas mentioned are integrated to implement an Analysis Module (AM) of SKC on a KC system. This module is in charge of processing explicit knowledge of the system in order to develop another latent and return it

to the system in a way that it can be used. For this reason, the module uses texts associated with the knowledge, the way the latter is structured and the way its components interact. As a result, the knowledge developed provides new access opportunities and interaction with the system and knowledge.

The knowledge tree of a KC node represents the common and shared understanding of the corresponding community on the domain dealt by the node. This tree may be considered a representation of an ontology underlying the domain (Gruber, 1993). Assignment of documents to topics in the knowledge tree involves semantic annotation of these documents within an ontology scope. This is the AM view of the knowledge tree node where it works. It is in this context, that one should be interested in the automatic annotation of documents -automatic assignment of documents to topics- (Kiryakov et al., 2004) or in mapping between ontologies -trees- with different nodes (Noy & Musen, 2002).

In AM text mining techniques that allow poorly structured textual data processing through the use of vectorial models are being used (Baeza & Ribeiro, 1999; Chang et al., 2001). These techniques are currently very popular, especially for their use in automatic indexing of the Web contents. In addition, AM uses analysis language processing (Carreras et al., 2004) appropriate for natural language processing. Application of these techniques in the field data recovery is not widely used because the computing effort of its use does not justify the benefits it provides in most common cases, where some of the texts compared are small and the repository to be dealt with is big -as it happens when using conventional search engines on the Web-. However, it seems that the situation may be different when comparing larger texts on moderate- sized repositories, which is the case we are concerned with and why we thought it would be a good idea to use this technique (Brants, 2004). There are other alternatives to this approach (Baeza & Ribeiro, 1999) that the prototype could be included in the future to contrast the results.

The ultimate aim of the module is to convert the result obtained into something useful to interact with the system and its contents. For this reason, it is essential to resolve problems related to the information filtering to be shown, typical of recommendation systems (Adomavicius & Tuzhilin, 2005), and with data visualization (Geroimenko & Chen, 2002).

To check the viability of the proposed approach several experiments have been performed with four KC nodes in learning activities carried out in the Universidad Autónoma de Madrid (Spain). The experimental results have shown evidences about how to take advantage of latent knowledge to enrich knowledge base and to facilitate the management task fulfilled by the system, the interaction among its entities and users' access to the contents that have been processed, among other interesting applications (Moreno-Llorena, 2008; Moreno-Llorena & Alamán, 2005; Moreno-Llorena et al., 2009a, 2009b). The enrichment of the proposed content seems to provide a very powerful support for automatic exchange of knowledge among knowledge management systems opening a way to the development of the latter on the semantic Web field (Berners-Lee, 2000).

2. SKC analysis module

The knowledge process that the Analysis Module (AM) produces for the Semantic KnowCat (SKC) (Moreno-Llorena, 2008; Moreno-Llorena & Alamán, 2005; Moreno-Llorena et al., 2009a, 2009b) system may be considered a digestion because its intention is to extract something new and assimilating by the system from the existing knowledge.

The AM works on the system knowledge base, depositing the result of its activity in the same repository, in a way in which both the system and users may use the module's contributions in a transparent way. The AM considers that the knowledge is formed by items. These items may be documents, topics, knowledge trees, nodes, users, etc.
With the new knowledge the system may improve the management it carries out in different ways, for instance, providing different views of the repository and new access services; simplifying users' classification of knowledge items in the system; or informing users about implicit relation between items, given the context of interaction.
Each knowledge item that is considered by the AM must have a description text associated, which may be assigned either manually or automatically; for the second option the module itself may deal with it on some occasions. The descriptive texts that are associated with the documents that SKC currently manages are the documents themselves, given that they contain textual information. In the case of the topics -that is a collection of documents-, are given by the descriptive texts of documents classified within themselves or of the subtopics they contain; although initially model texts that don´t necessarily have to be part of the system knowledge repository, may be used. Nodes are the same way in that they may be considered the root topics of a knowledge tree constituted by the topics and the documents included within it. Regarding users, several description texts may be associated with them, considering the documents or topics that, for instance, they provide or use frequently.
The AM carries out two fundamental tasks: on the one hand, it develops knowledge that is latent in the system; and on the other hand it incorporates it within the system itself in an explicit way in order to allow its exploitation. Implicit knowledge is found in relations that are established among the different knowledge items, for instance within the contents included or within interactions that ones and others establish. Explicit knowledge is incorporated into the system in its clear new state, describing the existing knowledge items, or in the form of new knowledge items that are added to the repository.
The link through the contents is established, in this approach, obtaining vectorial descriptors of the weight of the terms from text documents associated with the items. With these descriptors items may be compared, the distance that separates them may be determined and groups among them may be formed.
Associations based on the interaction between knowledge items are determined by analyzing how items relate among themselves. Like this, the way in which topics group documents and other topics in the knowledge tree of nodes may be considered, and how users provide documents to the system.
The knowledge incorporated into the system, as a result of the analysis, provides new repository exploitation opportunities. On one hand, the enriched knowledge items may be shown from new perspectives thanks to the new attributes. On the other hand, the items incorporated into the system by the knowledge assimilated by the latter, allow offering users different views of the repository and new services.

2.1 Linking by content

In our approach we have considered, initially, four types of knowledge items: nodes that are system instances in charge of the knowledge management about an area with the help of a user community; topics structured in the form of a knowledge tree that develop the different aspects of the main node topic; users that constitute the community that participates in the node; and documents that describe the different topics and are provided by the users, searched by them and which are the object of their consideration.

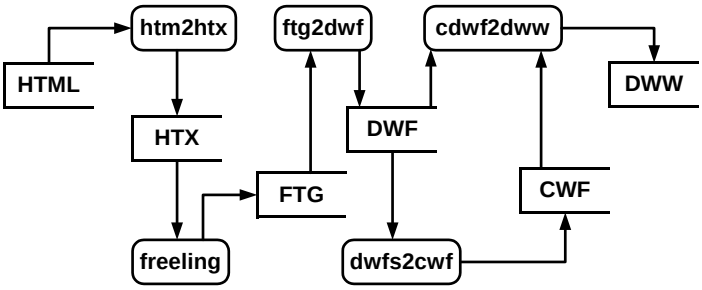


Fig. 1. Process obtaining weight of words vectors for knowledge items from HTML

It is possible to associate a text document with all the knowledge items that are considered in the system that describes them. These contributions may have several backgrounds. First, text associations to knowledge items may come from the nature of these items; for instance the documents used in experiments are textual type. Secondly, these associations may stem from the explicit relations of knowledge items with other items that already have associated texts. This occurs with topics that organize the documents, users that provide documents to the system or the node that contains both. Thirdly, text associations descriptive to the items may be inferred from more dynamic relations, like the one that is established between users and the documents that are more frequently visited by them or the ones they give their opinion about, or like the ones that are shown between documents referenced among them. Lastly, it is always possible to associate texts descriptive to the items that affect some aspects of utility, such as users' curriculums, their topics of interest, key words associated with documents or descriptions of topics. This case is completely general and may be applied to non textual documents, such as images, sound, etc.

Once a descriptive text has been associated with one of the items considered, it is necessary to put it in a way that it can be used as a comparison instrument. This is achieved by converting the text into a descriptor that shall be connected to the aspect it is referring to. For instance, if the text associated with a user describes the topics of interest for the latter, the corresponding descriptor shall refer to the users' preferences; but if the text describes the documents it itself has elaborated, the corresponding descriptor shall refer to its creative job. Like this, items shall be able to have as many descriptors as aspects of them are taken into account.

```
<html>
 <head>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
    <title>Arquitectura de Linux</title>
 </head>
 <body text="#330033" bgcolor="#CCCCCC" link="#3366FF" vlink="#000099" alink="#FF0000">
    <center><table COLS=2 WIDTH="73%" ><tr><td>
            <h1><font color="#CC0000">Arquitectura de </font></h1>
            <center><h1><font color="#CC0000">  Linux</font></h1></center></td>
        <td><img SRC="img4.jpg" height=114 width=102></td></tr>
     </table></center>
    <hr WIDTH="100%" color="#FFFFFF">
    <h2>Introducci&oacute;n</h2>
            Por arquitectura de Linux, y
    de cualquier otro sistema operativo en general, podemos entender que es
    la relaci&oacute;n estructurada que tienen los distintos componentes del
    sistema entre ellos, para cumplir su funci&oacute;n: proporcionar al usuario
    una <b>maquina virtual</b> sobre la cual trabajar.
...
```

Fig. 2. HTML file example

In our approach descriptors are weights of words vectors that may be used to determine similarities with other vectors of the same kind and thus relate the corresponding knowledge items (Baeza & Ribeiro, 1999; Chang et al., 2001). The process for obtaining these vectors starts from the texts associated with the items. As texts may be in different formats, they need to be treated in order to obtain their contents "naked" in the form of a flat text. In our approach text files in PDF and HTML (see Fig. 2) format have been considered, although both are transformed into flat text files before starting the process.

```
Arquitectura de Linux
Introducción Por arquitectura de Linux, y de cualquier otro sistema operativo en general,
podemos entender que es la relación estructurada que tienen los distintos componentes del
sistema entre ellos, para cumplir su función: proporcionar al usuario una maquina virtual
sobre la cual trabajar.
...
```

Fig. 3. HTX file example

After the text format has been eliminated- creating flat text files HTX (see Fig. 3)- the lemmas to which the terms refer to must be identified (obviating the grammatical forms in which they are shown) and to determine the grammatical categories to which they belong. With that, references are unified to concepts, the number of different words considered is reduced and the terms with no utility are identified.

In our approach we have used FreeLing language analysis tool (Carreras et al., 2004) that facilitates obtaining all the necessary information to achieve the previous objectives. FreeLing allows to analyse a text to identify the grammatical categories to which the words that form them belong to and to determine the lemmas to which these words correspond in a reference dictionary. When FreeLing cannot find an appropriate lemma to some word, it considers it a new lemma. With all this, the tool may establish the most probable morphologic interpretation of each word that integrates the text that shall be useful to determine a semantic approach of the latter. As a result of the analysis, FreeLing provides tagged version of the text (FTG files), indicating for each appearance of a word its original form together with the lemma and the corresponding morphologic interpretation that are considered more feasible. The Fig. 4 left shows a FTG file example, where for each row, the first string is the word original form, the second is the corresponding lemma, and the third is the encoding of the grammatical category.

```
...
y y CC
de de SPS00
cualquier cualquiera DI0CS0
otro otro DI0MS0
sistema sistema NCMS000
operativo operativo AQ0MS0
en en SPS00
general general AQ0CS0
podemos poder VMIP1P0
entender entender VMN0000
que que PR0CN000
es ser VSIP3S0
la el DA0FS0
relación relación NCFS000
estructurada estructurar VMP00SF
que que CS
tienen tener VMIP3P0
los el DA0MP0
distintos distinto DI0MP0
...
```

```
...
3   2 0.66 ejecutar VMG0000,VMN0000
4   2 0.66 first NCMS000
5   2 0.66 proceso NCMS000,NCMP000
6   1 0.33 Básicamente RG
7   1 0.33 Evidentemente RG
8   1 0.33 Paralelamente RG
9   1 0.33 TM_ms:200 Zu
10  1 0.33 abierto AQ0MSP
11  1 0.33 abrir VMN0000
12  1 0.33 abstraer VMN0000
13  1 0.33 acceder VMP00PM
14  1 0.33 acceso NCMS000
15  1 0.33 acelerar VMN0000
16  1 0.33 además RG
17  1 0.33 administración NP00000
18  1 0.33 administrador NCMS000
19  1 0.33 administrar VMIP3S0
20  1 0.33 affs NCMP000
...
```

Fig. 4. FTG (left) and DWF (right) file examples

The text that has been tagged using FreeLing is processed according to its grammatical categories, in order to completely eliminate entry words that are not considered relevant for the comparison of texts, such as determiners, conjunctions or prepositions. Tags and the original form of other entries are also eliminated. In this way the original text shall remain a sequence of lemmas that already exist in the reference dictionary or that have been minted from outstanding terms that do not appear in it. In this sequence entries for different forms of the same word in the original text appear as a repetition of the same lemmas. Each lemma included in this sequence may be ascribed to a semantic interest in order to contribute to the creation of a descriptor that is the objective of the process.

By counting the appearances of each term in the sequence of lemmas, it is possible to establish the frequency of each of them. In this way, word frequency files are generated for each text associated with a knowledge item (DWF files). DWF files include only one entry for each lemma that contains the corresponding identifier and its frequency, normalised with regard to the maximum appearances of other words taken into account in the document. The Fig. 4 right shows a DWF file example, where each row corresponds to a lemma and the columns are: identifier, appearances number, frequency normalised, lemma and grammatical categories.

Following a similar process to the one described –but working on a collection of representative texts for the general use of the language being used- a reference file is generated with the frequency of words in this collection (CWF file) that represents the frequency of words in the common use of the language (Baeza & Ribeiro, 1999). The document collection is processed as if it were the text associated with a knowledge item. So that the words found and their frequency is representative for the general use of the language, the collection must be broad enough and cover general themes. In our approach the 748 articles included in El País newspaper annuals from four different years that deal with the most outstanding events that took place during that period in the main field of information such as society, culture, sports and so on, have been used.

CWF files are similar to DWF; they include one entry for each lemma, with the identifier in question and its frequency coefficient for the inverse document. This frequency is logarithm to the base ten of the quotient of the total documents in collection N, between number nk of documents where the term appears (see For.1). This coefficient is an indicator frequency of the use of the term in the general use of the language that represents the collection and indicates the rareness of the latter. The Fig. 5 left shows a CWF file example, where each row corresponds to a lemma and the columns are: identifier, appearances number, inverse frequency normalised on document collection, inverse document frequency and lemma.

$$p_{k,i} = f_{k,i} \times fdi_k = f_{k,i} \times \log \frac{N}{n_k} \quad (1)$$

Considering the word frequency files of each knowledge item (DWF), and using the Word frequency file in the reference collection (CWF), a weight for each term is established in the text associated with the item. The weight of a word in a text represents the relevance of the term in it. A term is more characteristic of a text the more frequent it is in the corresponding text and the less frequent it is in the general use of the language in which it is written. Specifically, the weight $p_{k,i}$ of a term k in a document i is the result of the normalised frequency $f_{k,i}$ of the word k in text i, by the term frequency inverse document in the collection used as reference fdi_k (see For.1).

```
...
30 298 2.510 0.400 mayor
31 290 2.579 0.412 deber
32 283 2.643 0.422 mundo
33 278 2.691 0.430 poner
34 277 2.700 0.431 alguno
35 277 2.700 0.431 tres
36 270 2.770 0.443 político
37 268 2.791 0.446 público
38 266 2.812 0.449 Gobierno
39 263 2.844 0.454 económico
40 260 2.877 0.459 decir
41 260 2.877 0.459 día
42 259 2.888 0.461 general
43 253 2.957 0.471 menos
44 252 2.968 0.473 gran
45 251 2.980 0.474 caso
46 249 3.004 0.478 cada
47 248 3.016 0.479 medio
48 246 3.041 0.483 grande
...
```

```
...
1   1 0.333 5.000 1.667 Básicamente
2   1 0.333 2.397 0.799 Evidentemente
3   1 0.333 1.971 0.657 Paralelamente
4   1 0.333 5.000 1.667 TM_ms:200
5   1 0.333 5.000 1.667 abierto
6   1 0.333 0.546 0.182 abrir
7   1 0.333 5.000 1.667 abstraer
8   1 0.333 1.306 0.435 acceder
9   1 0.333 1.175 0.392 acceso
10  1 0.333 1.261 0.420 acelerar
11  1 0.333 0.554 0.185 además
12  1 0.333 1.427 0.476 administración
13  1 0.333 2.175 0.725 administrador
14  1 0.333 2.096 0.699 administrar
15  1 0.333 5.000 1.667 affs
16  1 0.333 5.000 1.667 aleatorio
17  1 0.333 0.806 0.269 algo
18  1 0.333 5.000 1.667 algoritmo
19  1 0.333 0.431 0.144 alguno
...
```

Fig. 5. CWF (left) and DWW (right) file examples.

The vector formed by the terms that appear in the text associated with a knowledge item and their respective weights, constitutes the resulting descriptor to the process, which is kept in the form of a file (DWW files). The Fig. 5 right shows a DWW file example, where each row corresponds to a lemma and the columns are: numeric identifier, appearances number, absolute and normalized frequency on document, inverse document frequency, word weight and lemma.

DWW shall be used to compare the items among one another, calculating the level of similarity among the weight of words vectors they represent. The similarity between two vectors is the measurement of the distance between them. In this approach we have considered the distance between vectors is estimated according to the cosine of the angle they form.

$$sim(v_i, v_j) = \frac{\vec{v}_i \bullet \vec{v}_j}{|\vec{v}_i| \times |\vec{v}_j|} = \frac{\sum_{k=1}^{t} p_{k,i} \times p_{k,j}}{\sqrt{\sum_{k=1}^{t} p_{k,i}^2} \times \sqrt{\sum_{k=1}^{t} p_{k,j}^2}} \quad (2)$$

Therefore, the similarity between two vectors v_i and v_j is the scalar product of the two vectors, broken up by the product of the respective modules. The scalar product of these vectors is calculated by the sum of the components of the product p_k in each of its t dimensions. A module vector is calculated by the sum of the squares of the components of a vector (see For.2).

```
...
          c0   c1   c2   c3   c4   c5   c6   c7   c8   c9   c10  c11  c12  c13  c14  c15
          T102 T103 T104 T105 T106 T107 T109 T110 T111 T112 T53  T54  T55  T56  T57  T58
f0   D100 0.08 0.07 0.04 0.06 0.00 0.16 0.03 0.05 0.01 0.04 0.06 0.04 0.05 0.04 0.02 0.12
f1   D101 0.05 0.05 0.01 0.03 0.00 0.00 0.03 0.01 0.02 0.03 0.04 0.01 0.03 0.03 0.02 0.04
f2   D102 0.07 0.09 0.13 0.13 0.00 0.00 0.10 0.03 0.06 0.11 0.12 0.13 0.16 0.11 0.12 0.21
f3   D105 0.12 0.09 0.05 0.17 0.00 0.00 0.13 0.05 0.12 0.11 0.14 0.05 0.22 0.50 0.20 0.15
f4   D109 0.08 0.13 0.14 0.14 0.00 0.00 0.12 0.08 0.09 0.11 0.12 0.14 0.18 0.13 0.10 0.17
f5   D111 0.12 0.12 0.16 0.17 0.00 0.00 0.11 0.05 0.08 0.11 0.14 0.16 0.22 0.17 0.15 0.19
f6   D113 0.00 0.02 0.00 0.00 0.00 0.85 0.00 0.01 0.00 0.00 0.03 0.00 0.00 0.03 0.01 0.00
f7   D114 0.14 0.19 0.07 0.11 0.00 0.01 0.13 0.09 0.10 0.11 0.14 0.07 0.12 0.12 0.08 0.14
f8   D115 0.02 0.02 0.00 0.03 0.00 0.00 0.00 0.00 0.00 0.00 0.02 0.00 0.00 0.02 0.00 0.03
f9   D116 0.14 0.19 0.07 0.11 0.00 0.00 0.12 0.08 0.11 0.11 0.13 0.07 0.14 0.09 0.08 0.16
f10  D123 0.15 0.21 0.15 0.11 0.00 0.00 0.13 0.08 0.10 0.12 0.11 0.15 0.13 0.13 0.09 0.15
...
```

Fig. 6. CDF file example.

The level of similarity between two vectors is a coefficient between zero and one. The closer the value is to the unit, the more similar vectors are, and the closer to zero, the less similar they shall be. The relation of similarity established between knowledge items is described with this coefficient. In our approach the knowledge items that exceed a specific threshold of similarity coefficient of the relation between both are considered to be related. Unfortunately, this threshold may not be established neither in a fixed way nor in a general way for all cases, given that depending on circumstances such as theme nodes or the nature of the documents taken into account, the election of its value may vary greatly.
The similarity between two set of DWW is summarized in a CDF file. The Fig. 6 shows a CDF file example, where rows f_i and columns c_i represent DWW files (respectively of documents and topics in this case) and he numbers the similarity sim(f_i,c_i) between them. In Fig.6, for instance, sim(f6,c5) (similarity between the document D113 and the topic T107) is 0.85, that is much higher than sim(f6,c10) (similarity between the document D113 and the topic T53), that is 0.03.

2.2 Linking by interaction

This kind of linking between knowledge items is established by analysing how these are related to each other. In our approach we have considered the way in which the documents and the topics within the knowledge tree are organised, and how users relate to the documents they provide to the system. The analysis follows a process that goes through three stages.
Firstly, the AM establishes the knowledge items included in the tree that need to be treated. The first time that a process is carried out in a node, all the tree knowledge items must be processed, but in consecutive processes only the items that have changed need to be treated or the ones linked to these. In our approach, for instance, changes in documents affect the topics of the branch of the knowledge tree where they are located, the node and the items related to ones and others in some way, but do not affect all the elements in the repository. In this selective process of the items it seems to be essential in systems with large knowledge bases or with an intense activity.
Secondly, the AM identifies the users responsible for the valid knowledge items in the knowledge tree. In this approach only the links between users and the valid documents that they have provided to the system are considered. The text associated with each user, according to the documents they provide, is the link to all the descriptive texts. Other kind of links similar to these could be treated in a similar way.
Lastly, the AM recovers the textual components that constitute the texts associated with the knowledge items through the Web. In our approach, we have started from text documents that are linked to them in a consubstantial way, in order to establish the texts associated with other items according to the relationship (above-mentioned) taken into account among them. We have used GNU Wget program (GNU Wget, 2011) in our approach to recover files that contain textual information corresponding to the different knowledge items and to integrate them –where there are more than one- to form the descriptive texts associated. These texts are usually the link to several files; for instance the text associated with a topic shall be made up of texts associated with each document and subtopic included.

2.3 Knowledge enrichment and its exploitation

As mentioned before, in our approach the knowledge developed, as a result of the analysis process, is incorporated into the system in an explicit way; either as descriptors that describe the pre-existing knowledge items or in the form of new knowledge items.

The descriptors added to the knowledge items provide new data to show hidden aspects of the elements they describe. For instance, the interest a particular item arouses may be suitable to make it stand out among the other items or to put all in order. In addition, the most characteristic terms that an item includes may result in an interesting reference to search information related to it in other information repositories.
In our approach, the knowledge specified by the analysis process is incorporated into the system in the form of a new knowledge element category that represents the relation among items of all kinds previously considered (documents, topics, users and nodes). The links incorporated this way in the repository provide the base to offer users new multidimensional views of the knowledge and new services to facilitate its exploitation. In particular, to demonstrate this proposal we have implemented an interactive view of the graph of a relation among knowledge items in the system (see Fig. 7 left), as well as a context sensitive recommendation service that provides reference items -of different kinds- related to the item the user is working with in each moment (see Fig. 7 right).

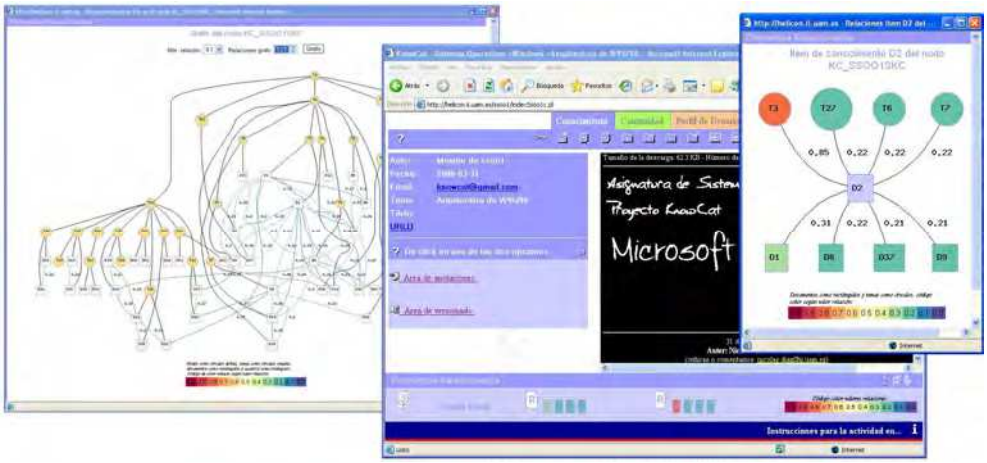


Fig. 7. Interactive view of the knowledge as graph related items (left window) and a context sensitive recommendation (inferior center window and right window).

The view in the form of a graph integrates the static relations established in the system with other dynamics that progress through time. Among the first we can find hierarchical links that join the topics of the knowledge tree or authorship links that connect users to the documents they provide to the knowledge base. Among the dynamic relations the derivation of the item's character present in the repository in each moment and the ones due to the interactions established among them as a result of the system activity may be mentioned. An example of this kind of view can be seen in the illustration above (see Fig. 7 left), where the topics are represented by orange circles, the documents by clear squares, the static relations by black lines and the dynamic relations between the items taken into

account are represented in the form of colour lines according to the level of similarity among the corresponding item vectors, included as a tag.
On the other hand, the recommendation service illustrates how to profit from the new knowledge to facilitate the use of the system and to make interaction with it more dynamic and attractive. In the illustration above (see Fig. 7 right) we can see an example of the window system showing a document and incorporating a recommendation panel on the bottom part, where representative icons of different kinds of knowledge items appear in warmer colours representing the high level of similarity among the vectors associated with the corresponding items. In addition, we can see in the window on top a representation in the form of a graph of the most important relationship that the document we are working with has with other items in the system. In this graph, as in the example of the view, the topics are represented by circles and the documents by parallelograms. However, in this case the colours of the figures represent the coefficient of similarity of the relations that link with the central knowledge item. Other services would be possible applying a similar approach, such as an assistant to locate documents in the most appropriate topic within the knowledge tree, or one to find experts in some topic or other users interested in the same subjects.
Both the view in the form of a graph and the recommendation service implemented allow navigation by mode knowledge different to the one the system allowed before making use of latent knowledge of the system. In both cases, the graphs have been generated by Graphviz (Gansner & North, 2000).

3. Experiments performed

To check the viability of these approaches, we have developed a prototype of the three elements shown that are part of the SKC system: analysis module (AM), graph visualizer for relation among knowledge items (KV) and context recommendation service (RS). They have all been incorporated into a KnowCat system (KC).
The prototype has allowed to perform several experiments with KC nodes, having been prepared for this during several years in teaching activities carried out in la Escuela Politécnica Superior de la Universidad Autónoma de Madrid. In particular, four KC nodes have been used: one node Operating Systems (OOSS); two Formal Languages and Automata Theory (FLAT); and one more Computer systems (CS).
The node Operating Systems is the result of the development of a list of topics on this subject carried out by the students during four consecutive years and which consists of a two level depth knowledge tree with over 20 topics and 350 documents.
The nodes FLAT organise different documents into two knowledge trees provided by the students during the academic year. Both nodes deal with the same subject, but in each of them the documents and the structure of the list of topics are different. Both trees have two levels, one node with 6 topics and 24 documents and the other with 12 topics and 50 documents.
Lastly, in node CS a topic about the corresponding knowledge area has been developed, hence the students from one subject have provided over 180 documents concerning around 40 different topics structured within a knowledge tree during an academic year.
The experiments carried out have been addressed to check the viability of automatic grouping of knowledge items using weight of words vectors assigned by the proposed

procedures. For this, three groups of experiments described in the following paragraphs have been performed.

3.1 Automatic grouping and classification experiments of knowledge items

In order to prove automatic grouping and classification of knowledge items three experiments have been carried out.

The first one starts from node KC on OOSS, where the two most successful documents for each item have been used to establish the weight of words vectors (WWV) for each one of them. Next, the levels of similarity between the WWV for the remaining documents and the WWV for the topics previously mentioned have been established. With these data a graph (see Fig. 8) has been obtained, where each row of points corresponds to one topic and each column to a document. The levels of similarity between documents and topics have been represented by colour points, the higher the value of coefficient, the lighter the colours. The documents have been organised into topics which students had initially classified manually.

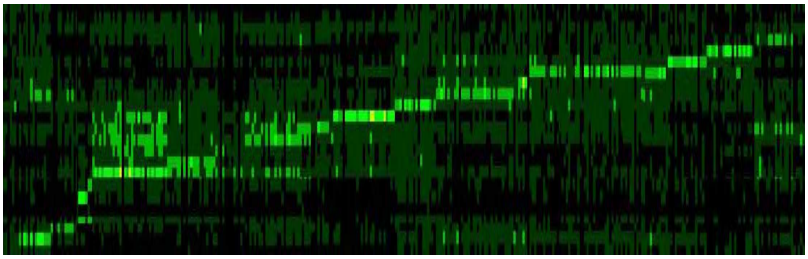


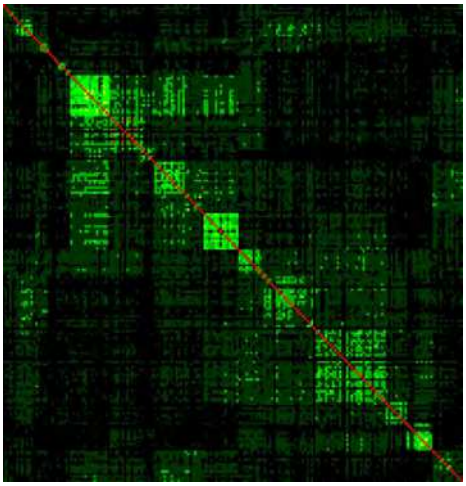
Fig. 8. Automatic classification of documents (rows) by topics (columns) in node KC on OOSS

As a result of the first experiment, in the graph we can see how the highest similarity values -lighter points- are aligned mainly in the rows that correspond to the topics in which they were classified manually. This indicates that automatic classification matches with manual classification in most cases. By analysing the anomalies a posteriori it can be seen that they correspond to ambiguous topics in which heterogeneous documents had been classified.

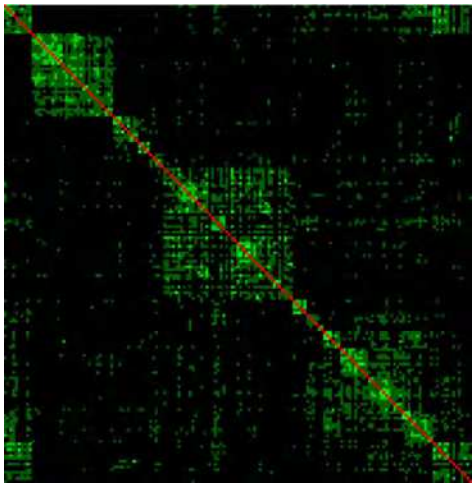
The second experiment compares the WWV of the documents of the original node OOSS between them. In the graph (see Fig. 9 left) each row of points -vertical and horizontal- correspond to a document and these appear organised into the topics in which they were classified manually. Like in the previous case, the levels of similarity among elements have been represented by colour points, the higher the value of coefficient, the lighter the colours.

As can be seen in the graph corresponding to this second experiment, the highest degrees of similarity are grouped into blocks around the diagonal. Under the conditions shown, this indicates that most documents are more similar to each other when they deal with the same subjects. However, in this case it is interesting to see how the light points outside the groups of the diagonal appear in bands that show how relationships among documents of similar topics are established. In the third experiment performed, the results are similar to the previous one where compared, in the same conditions, the WWV of the documents of a KC node on CS, as can be seen in the corresponding graph (see Fig. 9 right). In this case, as the

number of topics is higher and the number of documents per topic is lower, groupings appear like smaller light blocks.



(a) OOSS KC node documents similarity

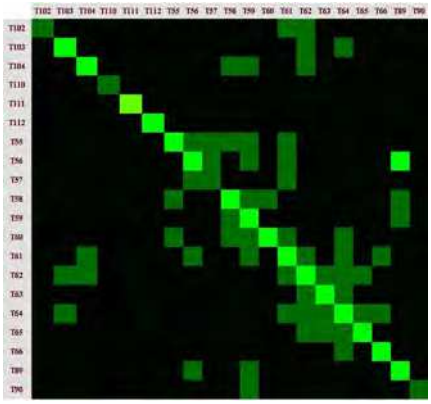


(b) CS KC node documents similarity

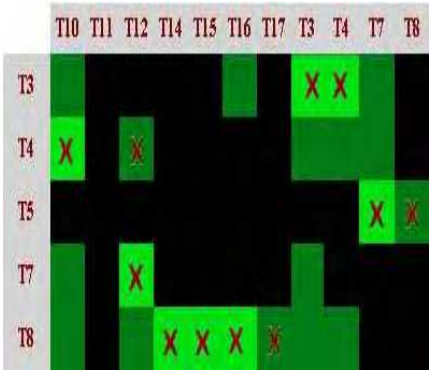
Fig. 9. Automatic grouping of documents by topics of knowledge area

3.2 Mapping experiments among knowledge trees

In order to prove mapping among the knowledge trees that represent the ontologies which are taken into account in each node of the system, two experiments have been carried out.



(a) OOSS KC nodes topics mapping



(b) FLAT KC nodes topics mapping

Fig. 10. Mapping between topics of KnowCat nodes

In the first experiment of this series, again, we start up with KC node on OOSS, although in this case two new nodes have been created from it with knowledge trees equal to the original. The documents of the initial node have been divided up into the recently created

ones, so that every couple of new homologous topics has a similar number of different documents, but relevantly similar. Later the WWV of the topics of the new nodes have been calculated and have been compared with each other. A graph (see Fig. 10 left) has been produced with the values of similarity obtained, where the lines of colour blocks correspond to the topics of one node and the columns to the other. The topics have been organised into both dimensions in order that the homologous topics are in the same position in the corresponding entries of the table. Like in previous graphs, the highest values of similarity are represented by the lighter colours.

As can be seen in the image of this first experiment, the highest grades of similarity -blocks of light colour- are over the diagonal in almost every case. With the proposed approach, this means that it is possible to identify the branches of the knowledge trees that contain documents dealing with the same topics.

For the second experiment two KC nodes on FLAT that have different trees to organise the knowledge have been used. Again the WWV of the topics have been calculated from the documents included within them and the grades of similarity have been calculated from the topics of different nodes comparing their corresponding vectors. The result is shown in a graph (see Fig. 10 right) where the topics of one node are in the axis of abscissa and the other in the organised axis. As on other occasions, the grade levels of similarity are shown in colour blocks, where again, the higher the value of coefficient, the lighter the colours. In this case, the pair of topics that are considered linked to each other through their contents by means of a manual analysis by an expert on the subject have been marked with a cross.

As a result of this second experiment, it can be seen that most of the associations made by an expert fit in over light colour blocks and that every light block is found in topic pairs associated by the expert. Therefore, it is possible to identify the proposed procedure and the topics that deal with related issues in different knowledge trees automatically.

3.3 Automatic association experiment among knowledge nodes

Starting from the documents included in five KC nodes, the one belonging to CS used in the first group of experiments, the two OOSS prepared for the previous group and the two FLAT used in the same group, a WWV has been established for each of them. In every case, the documents included in the nodes are different. By comparing these WWV a graph (see Fig. 11) has been obtained, in which each line of blocks, vertical and horizontal,

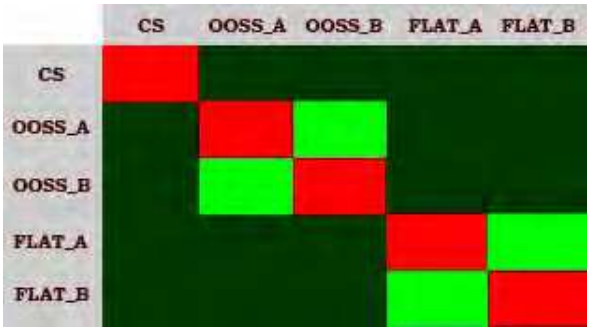


Fig. 11. Grouping of KC nodes per topics

corresponds to a node. Like in previous diagrams, the lighter colours represent a greater similarity.
In the graph we can see that the level of similarity among the WWV of the nodes that deal with the same topics are high compared with the ones obtained where comparing the node vectors on different topics. This means -using this technique- that it is possible to identify nodes that deal with similar contents and to distinguish them from others on different subjects.

4. Conclusions and future projects

Semantic KnowCat (SKC) is a prototype developed on KnowCat to investigate solutions to information overload in ICT-based systems, using knowledge management systems as a model. SKC uses for this purpose some hidden aspects of such systems, as the residual energy of their activity, and properties of both the elements and the activities involved.
The process of the digestion of knowledge proposed seems to be able to specify latent knowledge in a knowledge management field, which may be useful to facilitate the management task fulfilled by the system, the interaction among its entities and users' access to the contents that have been processed, among other interesting applications (Moreno-Llorena, 2008; Moreno-Llorena & Alamán, 2005; Moreno-Llorena et al., 2009a, 2009b). The enrichment of the proposed content seems to provide a very powerful support for automatic exchange of knowledge among knowledge management systems opening a way to the development of the latter on the semantic Web field (Berners-Lee, 2000).
However, the threshold found in the levels of similarity to consider the similar knowledge items is low and higher values are unlikely to appear. In almost every case taken into account most of the items having similarity over 0.3 are related to each other for their contents and the ones that aren't have minor levels, although some objectively related do not reach that value. In some cases the threshold is even lower, between 0.2 and 0.3. It would be highly desirable that the level of similarity would mark more clearly the space between items with different contents and would clarify the similarity between those that have similar contents.
With all this, it is considered highly interesting to continue advancing in an open line of work, paying special attention to specification and contrast of the level of similarity, and searching integration of content analysis proposed with the one for interaction of users (Moreno-Llorena et al., 2009a) and with automatic interaction among nodes (Moreno-Llorena et al., 2009b).

5. Acknowledgements

The system KnowCat and Semantic KnowCat prototype were partially financed by the Spanish Ministry of Science and Technology, project codes TIN2004-03/40 and TSI2005-08225-C07-06. From 2003 to 2007, the system KnowCat was exploited within the frame of four teaching innovation projects (TIP) financed by Universidad Autónoma de Madrid. In the scope of these TIPs, the undergraduate fellow student Javier Hidalgo has collaborated especially on the work shown here. The current research has been partially financed by

Spanish National Plan of R+D, project code TIN2008-02081/TIN, and by the CAM (Autonomous Community of Madrid), project code S2009/TIC-1650.

6. References

Adomavicius, G. & Tuzhilin, A. (2005). *Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions*, In : IEEE Transactions on Knowledge and Data Engineering No. 17, pp. 734-749, 2005

Alamán, X. & Cobos, R. (1999). *KnowCat: a Web Application for Knowledge Organization*, LNCS 1727. Eds. Chen, P.P. et al. Springer, 348-359. 1999.

Baeza, R., Ribeiro, B. (1999). *Modern Information Retrieval*. Addison Wesley, 1999.

Berners-Lee, T. (2000). *Semantic Web - XML2000*. Available from http://www.w3.org/2000/Talks/1206-xml2k-tbl

Brants T. (2004). *Natural Language Processing in Information Retrieval*. In proceedings of CLIN 2004 Antwerp, Belgium, 1–13. 2004.

Carreras, X., Chao, I., Padró L. & Padró M. (2004). *FreeLing: An Open-Source Suite of Language Analyzers*. In proceedings of LREC 2004 Lisbon, Portugal. 2004.

Chang, G., Healey, M., McHugh, J., Wang, J. (2001). *Mining the World Wide Web: An introduction search approach*. Kluwer, 2001.

Cobos, R. (2003). *Mechanisms for the Crystallisation of Knowledge, a proposal using a collaborative system*. Doctoral dissertation. Universidad Autónoma de Madrid.

Cobos, R., Pifarré, M., (2008). *Collaborative knowledge construction in the web supported by the KnowCat system*. Computers & Education, Vol.50, No. 3, (April 2009), pp. 962-978, ISSN 0360-1315.

Gansner, E., North, S. (2000). *An open graph visualization system and its applications to software engineering*. Software – Practice and Experience, 30:1203–1233, 2000.

Geroimenko, V., Chen, C. (2002). *Visualizing the Semantic Web*. Springer, 2002.

Gruber, T. R. (1993). A *Translation Approach to Portable Ontology Specifications*. Knowledge Acquisition, 5(2), pp. 199-220, 1993.

Kiryakov, A., Popov, B., Terziev, I., Manov, and D., Ognyanoff, D. (2004). *Semantic Annotation, Indexing, and Retrieval*. Journal of Web Sematics 2, Issue 1, Elsevier 49-79, 2004.

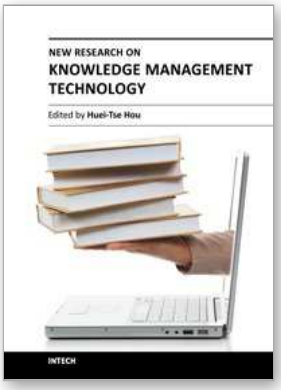
Moreno-Llorena, J. (2008). *Collaborative Knowledge Management By Means Semantic Information*. Doctoral dissertation. Universidad Autónoma de Madrid.

Moreno-Llorena, J., Alamán, X. (2005). *A Proposal of Design for a Collaborative Knowledge Management System by means of Semantic Information*. In: Navarro-Prieto, R. et al., HCI related papers of Interacción 2004, Springer, Dordrecht, The Netherlands,. 307-319, 2005.

Moreno-Llorena, J., Alamán, X., Cobos, R. (2009a). *Modeling of User Interest Based on Its Interaction with a Collaborative Knowledge Management System*. Lecture Notes in Computer Science (ISSN: 0302-9743, impreso y 1611-3349, en línea; ISBN: 978-3-642-02579-2), 330-339, Springer Berlin / Heidelberg. 2009.

Moreno-Llorena, J., Alamán, X., Cobos, R.(2009b). *Establishment and Maintenance of a Knowledge Network by Means of Agents and Implicit Data* . Data Mining and Multi-agent Integration (ISBN: 978-1-4419-0523-9, impreso y 978-1-4419-0522-2, en línea),155-166, Springer US. 2009.

Noy, N. F., Musen, M. A. (2002). Evaluating *Ontology-Mapping Tools: Requeriments and Experience*. In EKAW02 Workshop (WS1) Sep 2002.
GNU Wget (2011). Available from http://www.gnu.org/software/wget



New Research on Knowledge Management Technology

Edited by Dr. Huei Tse Hou

ISBN 978-953-51-0074-4
Hard cover, 228 pages
Publisher InTech
Published online 24, February, 2012
Published in print edition February, 2012

Due to the development of mobile and Web 2.0 technology, knowledge transfer, storage and retrieval have become much more rapid. In recent years, there have been more and more new and interesting findings in the research field of knowledge management. This book aims to introduce readers to the recent research topics, it is titled "New Research on Knowledge Management Technology" and includes 13 chapters. In this book, new KM technologies and systems are proposed, the applications and potential of all KM technologies are explored and discussed. It is expected that this book provides relevant information about new research trends in comprehensive and novel knowledge management studies, and that it serves as an important resource for researchers, teachers and students, and for the development of practices in the knowledge management field.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821