Knowledge Retrieval and Personalization in Virtual Enterprises

Claudio BIANCALANA
Department of Computer Science and Automation
Artificial Intelligence Laboratory, Roma Tre University
Via della Vasca Navale, 79 – Rome, 00146, Italy

Fabio GASPARETTI
Department of Computer Science and Automation
Artificial Intelligence Laboratory, Roma Tre University
Via della Vasca Navale, 79 – Rome, 00146, Italy

Alessandro MICARELLI
Department of Computer Science and Automation
Artificial Intelligence Laboratory, Roma Tre University
Via della Vasca Navale, 79 – Rome, 00146, Italy

A B S T R A C T

Each business company collects, produces and exploits for its activities and goals large amounts of information. Most of the times this knowledge makes the intellectual capital for creating value and innovation. Knowledge management (KM) systems aim at manipulating knowledge by storing and redistributing corporate information that are acquired from the organization's members. In this context, Virtual Enterprises (VE) plays a crucial role as not permanent alliances of enterprises joined together to share resources and skills in order to better respond to business opportunities. The representation and retrieval of distributed knowledge is an important feature that information systems must provide in order to obtain advantages from this kind of enterprises. KEEN[1] (Knowledge-based Extended Enterprise) is a research project for developing a system able to extract and let different business companies access to collective knowledge required to achieve particular shared goals. In this paper, we report the most important features of this system, especially in the context of distributed knowledge representation and retrieval.

Keywords: KEEN, knowledge management, retrieval, elicitation, information, used modeling.

1. INTRODUCTION

Knowledge management (KM) [1] has been recognized as a fundamental asset in the global market place. Companies' know-how and the accumulated knowledge must be collected and made easily accessible to enhance the efficiency and effectiveness of knowledge-intensive work processes and competitive advantages. Improve the capitalization on existing knowledge assets facilitates the creation of new knowledge, profit returns and innovation.

In spite of the apparent simplicity of the term, there are not clear definitions and classifications of knowledge management.

Some experts described the purpose of KM essentially as a document management system. Other experts prefer to focus on the process of handling unstructured knowledge, or more in general, technical and organizational initiatives to manage structured and unstructured knowledge in order to store and reuse the internal company's knowledge.

In this context, Knowledge Management Systems (KMS) are defined as Information Technology-based systems to support and enhance the organizational processes of knowledge creation, storage/retrieval, transfer, and application [1].

The process to capture and store knowledge in ad-hoc repositories to be able to quickly retrieve information according to the user needs and goals plays a predominant role in KMS.

Knowledge engineers might help to extract knowledge by *elicitation* activities. Direct elicitation methods such as storytelling, interviewing and question answering provide required information directly from domain experts interviewed by knowledge engineers that know what knowledge will be elicited. In indirect elicitation methods, information is not directly obtained from domain experts but there is a further step where knowledge engineers are involved in the analysis of the results of elicitation sessions. This second approach is particularly useful when the knowledge engineers have not fully explored the current domain or some knowledge has been ignored during the direct elicitation sessions. In these circumstances, indirect methods help obtaining information that cannot be easily expressed directly.

While elicitation methods are useful for capturing knowledge from users or groups working on particular tasks, the large amount of information stored in paper documents or databases are paramount sources of rich knowledge to exploit during company's activities. Examples of information maintained at the work group level or beyond includes: reports, procedures, pictures, video tapes, technical standards and databases.

Many people believe that semi-structured or structured information play an important role in a company's knowledge management [12]. Structured information is intended to unambiguous and explicitly represents concepts in formats that describes each necessary attribute and property, e.g., relational database table, while unstructured information usually requires a human interpretation in order to extract its intended meaning, e.g., natural language documents, audio, still images, and video. Structured information allows users to find, share and integrate information with more precision, also with the support of software agents [14].

Nevertheless, unstructured information covers around 80 percent of all corporate information [20], and even several public available digital libraries that can be employed for business activities such as the Web, are mostly composed of unstructured information.

If we consider Virtual Enterprises (VE), where strategic alliances amongst non-competing companies are settled for the accomplishment of specific goals, there importance of sharing and recovery of useful structured/unstructured information among partners becomes even more crucial

A company may be not completely aware of the knowledge

1 Project funded by the Italian Ministry of Education, University and Research (prot. N° 4418/ICT)

stored by the partners of the VE. Bringing collectively available complementary competencies and resources is one of the major achievements of a VE.
In this paper we describe KEEN (Knowledge-based Extended Enterprise), an under development KMS able to retrieve and share knowledge shared by different companies grouped in a VE.
One on the most important features of the proposed system is the ability to manage different kinds of information representations and degree of formality, i.e., structured, semi-structured or unstructured, providing an uniform access to several different information sources. Instead of long and costly elicitation processes to manually extract and annotate knowledge, KEEN uses machine learning and information extraction techniques in order to draw knowledge from the information stored in the companies' internal digital libraries and map it into the enterprise ontology. User modeling is employed to adapt the interaction with the information system in order to personalize the results of humans' information seeking activities.
In the following section, we introduce some works related to knowledge management in distributed environments. Section 3 and 4 describe the techniques used for indexing and retrieval of knowledge in KEEN. Section 5 briefly shows the KEEN architecture.

2. RELATED WORK

In recent years, knowledge management has become a focus of attention for many organizations. Knowledge is considered to be the key source for sustainable competitive advantage [21][7]. Therefore, Knowledge management is aimed at locating, capturing, transferring, sharing and creating knowledge within and across organizations [29][30]. It will be clear that conceptual modeling, as developed within the field of KBS construction, provide key techniques for knowledge management [10][27][24].
Implementing a KM system is not a simple tasks due to several issues noted [19]:

- Technology issues: KM involves software technology, but it is not always simple or even possible to integrate all the different subsystems and tools to achieve the level of sharing that was planned.
- Inadequate security: While the idea behind KM is to share knowledge, it is important not to share knowledge assets with the wrong audience (e.g., competitors and former employees).
- Individual issues: Employees do not have time to input knowledge or do not want to give away their knowledge.
- Lack of standards in knowledge representation

A characteristic that turns out to be an advantage over other industries in terms of managing intellectual capital is that artifacts (documents) are already captured in electronic form and can easily be stored and shared. In addition, software engineers often have a friendly attitude towards using new technology. This means that a software organization that implements a knowledge management system could have a good chance to succeed with this mission. However, this remains a challenging task because a knowledge management system is more than just technology. There are only a few published works about initiatives to manage knowledge in software organizations, but all of them talk about the difficulty of achieving employees' acceptance and implementing the KM system in a way that maximizes the help provided to its users [24][25][3][17][28].

3. KNOWLEDGE INDEXING

Before facing the problem of knowledge retrieval, it is essential to analyze how the system indexes the available information, that is, which representation has been chosen to guarantee an efficient and effective retrieval phase. In particular, the requirements are twofold: it is essential that knowledge is quickly retrieved by users, and this knowledge accurately satisfies the users' information needs in terms of high precision. An indexing system for business companies must also be able to deal with different kinds of information representations, from unstructured documents based on natural language to ontology-based knowledge and relational databases. Moreover, it should provide a comprehensive and homogenous human-computer interface for knowledge retrieval.

In order to provide the aforementioned prerequisite, it is necessary to consider different types of information and the degrees of information "richness". Information based on *ontological standards*, for instance, expresses relationships between typically non-structured information, e.g., natural language text, and meta-data. These meta-data usually state features or classes related to given peaces of information. A typical example is the association between a document and one particular category in a predefined taxonomy.
As for information stored in *databases*, we have an underlying relational model that clearly states the semantic meaning of each peace of informative unit, e.g. price, address, location, etc., and therefore facilitate the interpretation/recovery process.
In order to define a unique representation that deals with the different types of available information, i.e., natural language, ontology-based and databases, we must define a subset of shared features that is possible to generalize, and automatic or semi-automatic methods and techniques for translating information from one of these representations to the internal one.
Ontology languages capture high-quality relationships and meta-data content that enable logic-based agents to interpret and take decisions autonomously [14][31]. The Semantic Web extends this vision to the Internet domain, where information is no longer based on HTML, but on semantic standards like OWL (Web Ontology Language), or preliminary standards like DAML+OIL. Nevertheless, research on these representations is not completed, and the logical engine able to automatically interpreting such information within single or multi-agent systems with traditional computational resources is yet to be defined. Moreover, much of the current available information is written by humans in natural language so additional effort is required to translate this information in the new ontology-based languages. Translating large amount of non-structured information into new formalisms is not an activity that can be accomplished manually. Research to develop methods and techniques for this goal has not reached completely satisfying results.
As for *unstructured* information based on natural language poses many problems during the indexing, but it is also particularly problematic to retrieve it efficiently. The well-known vocabulary problem [9] for instance, points out further issues in terms of synonymity and polisemy of words that do not allow users to express univocally their information needs.

Internal representation of knowledge
As previously stated, the proposed internal representation of knowledge defines some common features shared among the three kinds of information briefly described. This sort of

intermediate representation consists of traditional non-structured information with associated meta-information related to concepts of a taxonomy of the business domain for the given virtual enterprise (see **Errore. L'origine riferimento non è stata trovata.**). In a few words, each informative unit is classified in a subset of categories from a simplified ontology.
Such meta-information can be exploited both in the retrieval phase, reducing possible ambiguities in the processed information, and to re-organize the knowledge in more efficient ways for further user search activities, e.g., online hierarchical clustering [12].

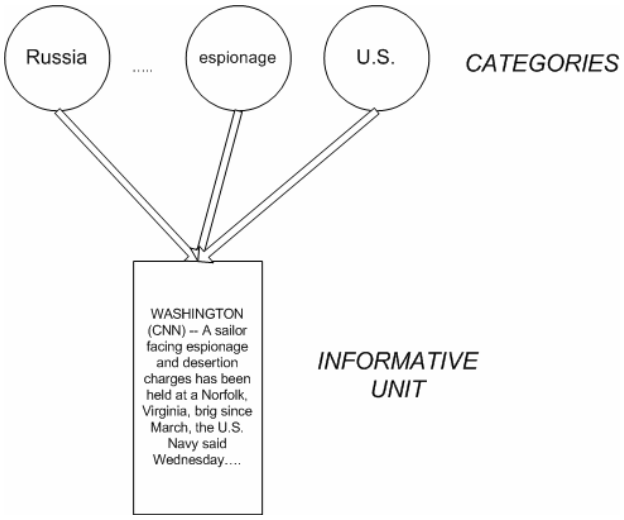

Figure 1: Each informative uint, e.g., a document or a segment of it, is associated with one or more categories in a given taxonomy

One further advantage of such a representation is the chance to exploit traditional and well-known indexing and retrieval techniques developed in the Information Retrieval field, as search engines based on the Vector space model [23]. Such systems guarantee quick response time thanks to data structures appropriately studied for efficiently memorizing the input documents. Additional information can be easily indexed and retrieved together with the original data in appropriate fields that can be used during the recovery process.
Even though the proposed representation simplifies the stored meta-data, i.e., there are no relationships between informative units such as IS-A or HAS-A relations in ontology-based languages; the burden is now on populating the internal knowledge base given the available information. In other words we have to define techniques and methodologies to transform the information represented in one of the three-abovementioned typologies, into the proposed intermediate representation.

Structured Information (databases and ontologies)
Information based on ontological standards does not pose relevant issues. In this case, the source is based on a rich language while our internal representation simplifies some features, such as the kinds of relations between the concepts. In our representation we have relations <u,C>; where *u* is the informative unit and *C* is a set of categories in the given taxonomy related to the concept *u*. We only have IS-A relationships so it is not hard to extract them from the initial ontology. The selection of the most important concepts from the initial ontology is the only task that knowledge experts have to perform before the population of the internal taxonomy.
An important issue to be tackled is when there are more ontologies present within the same virtual enterprise, i.e., different companies own different ontologies. In this case, there are two choices: we can merge the ontologies with automatic techniques that try to find correspondences between them, or it is possible to manually match pairs of entities that represent the same concept from the two ontologies. Since efficient techniques for ontology matching is not available due to several difficulties in solving automatically and efficiently such problems, we decided to let the knowledge experts to manually associate correspondences between identical concepts belonging to different ontologies.

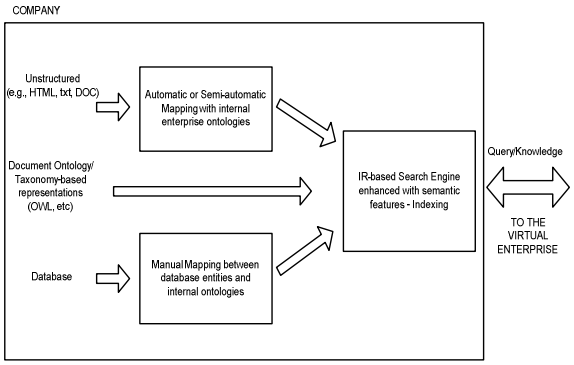

Figure 2: KEEN Indexing process of the three kinds of available information

As for data stored in relational models, i.e., databases systems, we usually have instances that adhere to formal and static schemes. These schemes formally describe the meaning of the stored information, e.g., LAST-NAME, ADDRESS, PHONE, etc.. Therefore, we can easily draw associations between database fields that are useful to store in the knowledge-base and taxonomy entities. If we assume that these schemes are static and that the useful information is store in a limited number of fields, the mapping between the database schemes and the internal taxonomy can be defined manually by knowledge experts.

Unstructured Information
Considering the current amount of unstructured information available within companies, the problem of making such information accessible to users is probably the most important issue to address in our knowledge system. In particular, having chosen a particular representation for virtual enterprise's domains, it is necessary to find a technique that allows to autonomously process the unstructured information and populate the internal knowledge base.
The scheme shown in Figure 3 shows the relevant technologies that allow such indexing process, that is assigning meta-information to unstructured data.

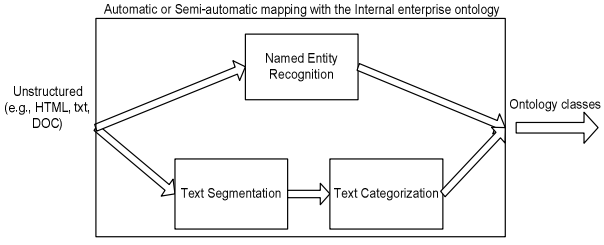

Figure 3: KEEN Indexing sub-process for unstructured information

The input is informative objects, typically text documents, reports, hypertext pages, emails, etc.. These objects are processed in two different flows:

- Named Entity extraction
- Text segmentation and categorization

Given an informative object, we initially locate and classify atomic elements in text into predefined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages, etc. We used Named Entity recognition (NER) techniques based on linguistic grammar-based techniques, statistical models and dictionaries (or gazetteers). NER is a well-known research field, subtask of information extraction, which does not focus on semantic interpretation of languages but on more practical and easier goals, obtaining excellent results in terms of precision of results. The output of NER, that is, the proper nouns of companies, persons, etc. is used to increase the weight of these entities during the indexing/retrieval steps.

A further step we follow to process the input information is text segmentation. In a few words, a given document is divided into sequences of words or other similar meaningful units that are separately stored in the knowledge base. This step is useful each time we have long documents, such reports or e-books which cover several different topics, i.e., categories in the internal taxonomy. In order to increase the precision of retrieval, it is better to split them into meaningful coherent regions. Our segmentation algorithm is based on the Choi's work [4]. It performs three steps:

- Tokenization
- Lexical score determination
- Boundary identification

Basically, after having tokenized the document into a sequence of tokens, we draw a similarity measure to analyze the semantic coherence among contiguous text regions. Finally, we determine the boundaries each time we have relevant variations of the semantic coherence measure.
The last process we perform on unstructured information is text categorization. After having recognized semantically coherent information units and, for each unit, its relevant entities, we assign a subset of taxonomy's categories to the unit by means of text categorization techniques [26]. In particular, the categorization is based on the Support Vector Machine (SVM) algorithm [15][16]. For the training phase, we use a subset of documents already categorized by the knowledge expert, performing an ad-hoc feature selection that exploits also the aforementioned NER module to draw more weight to terms that correspond to relevant semantic classes, e.g., proper names, locations, etc. The categorization output is a tuple of couples $<c_i,\alpha_i>$, where for each category c_i of the taxonomy we have a value α_i between [0,1] that represents the degree of relatedness of the input document to the class. This information is stored along with the document in the knowledge base and it will be used during the retrieval and personalization, as described in the following sections.

4. KNOWLEDGE QUERYING

The most popular paradigm for querying a textual database is to submit short queries. Users express their information needs through a small set of keywords that must be present in the retrieved documents. The retrieval system shows an ordered list of references, based on matching algorithms that assign a relevance weight to each indexed document.
In our knowledge base, along with each document, we have a list of assigned categories referenced in the internal taxonomy. If we want to exploit this information, the query should include one or more categories that users are interested in. We named this enhanced user needs, semantic queries. In order to let users specify a set of categories, we have developed a GUI that starting from a generic query shows a provisional result set.
The result set is clustered in groups that are characterized by having documents with similar categories. The user is able to see and choice the groups of his/her interests. Afterwards, the system adapts the initial query adding the chosen categories, updating the results sets filtering out the documents that have not been tagged with those categories. This approach has some sort of similarities with the Scatter/Gather system [22] and other online clustering systems [4][8]. Of course, in our GUI users are also allowed to directly choose a subset of categories of the taxonomy and see the most relevant results for each category, and formulating a query after having analyzed some informative units. This approach helps users that start an information seeking session without any clue about the keywords to specify during the search.
In order to retrieve the most important documents given a generic query, that is, without categories, the system is also able to perform an ad-hoc comparison between the query q and each category c_i. The comparison is based on the cosine similarity [23]:

$$sim(q,c) = \cos(\vec{q},\vec{c}) = \frac{\vec{q}\cdot\vec{c}}{|\vec{q}|\,|\vec{c}|}$$

where the representation of categories and queries is based on a Vector space model. For each category, we extract the whole content of the included documents. This set of keywords is weighted according to the TFxIDF weighting scheme. In this way, a vector v is assigned to a generic category c_i, and we are able to order the set of categories C by means of the cosine similarity, showing to users the most related categories given a query q. We summarize the weighting steps as follow:

1. For each document in the taxonomy's category c_i we extract the textual content
2. The content is tokenized and filtered through a stop-list of words
3. We draw nouns and entities by a Part-of-speech Tagger (POST) and the previously described NER
4. We perform a TFxIDF weighting on the retrieved set of terms obtaining a set of word-weight couples $\{(t_1,w_1)(t_2,w_2),\ldots(t_N,w_N)\}$ representing the vector of the category c_i

Query Disambiguation and User Modeling
One of the most important problems that occurs during querying a corpus of textual documents is the choice of the right keywords for the retrieval. As we already pointed out, synonymity (two words that express the same meaning) and polisemy (different meanings expressed by one word) of natual language [9] have the chance to decrease respectively the recall and precision of the retrieval process.
For that reason, we have included a user modeling component to represent the needs of the users. This component is involved during the querying in order to help disambiguating the meaning of the query's terms.

User modeling is usually employed to infer users' needs by considering a users' background, actions, and queries when they interact to information systems.

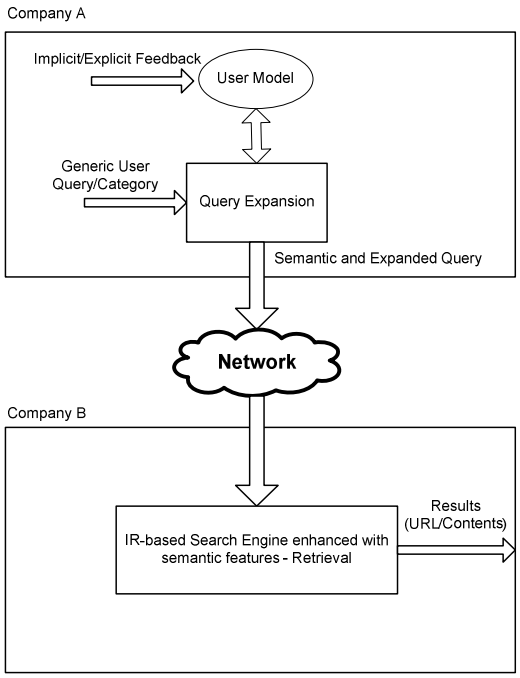

Figure 4: Personalization of Search Results

The proposed user modeling is based on a concept network paradigm [11] instantiated on the taxonomy of the enterprise's domain. Concept networks are usually employed as a form of knowledge representation. They consist of graphs of concepts, and edges, which represent semantic relations between these concepts. We use concept networks to weight which concepts the users are more interested in, that is, concepts related to the user needs. In our first prototype, the relations between concepts are not considered.
There have been proposed several methodologies to draw concept weights. We summarize our weighting scheme in the following steps:

5. For each user we keep track of viewed and edited documents, e.g., Words docs, Web pages, reports, emails.
6. From the collected documents we extract the textual content
7. We give the whole content as input to the categorization module described in the previous section
8. We draw a set of concept-weight couples $\{(c_1,w_1)(c_2,w_2),\ldots(c_M,w_M)\}$ related to the given taxonomy and we store it in the user model

Basically, we employ implicit feedback techniques [18] to retrieve data for user training, assuming the browsed and edited documents are related to the needs of the user in a company.
Finally, a *renting* process allows to reduce the weight of concepts that have not been recently covered by the user. In this way, we are able to ignore no-longer interesting concepts, or concepts that have been erroneously included in the user model.

As already stated, users can choose a set of categories before submitting a query. If they do not know the categories of interest beforehand, it is possible to navigate the search results selecting the categories assigned by the system to each document, refining the query. In this case, the system exploits the user model to disambiguate the initial query.

We use a query expansion technique to modify the initial query adding additional keywords identified from the most important categories C' in the user model. Briefly we extract the keywords with the algorithm described in the previous section (see "Semantic Querying") that are included in the expanded query according to the weight of the category in the user model. We summarized the query expansion as follow:

1. Set the vector Q' to 0
2. We build a set of word-weight couples $\{(t_1,w_1)(t_2,w_2),\ldots(t_N,w_N)\}$ representing the vector of a generic category c_i as described in the previous section
3. For the category c_i we retrieve the subset T_{ci} of word-weight couples that gain highest weight
4. We retrieve the weight w_{ci} of the category c_i in the user model
5. We perform an inner product between the vector T_{ci} and the weight w_{ci} adding the obtained vector to Q'
6. We iterate the points 2-5 through each category c_i of the taxonomy

The output vector Q' represents the expanded query that is submitted to the system.
For example, if we have the user model's category *O.S. – Linux* weighted 0.7, and the term *kernel* extracted from the same category has weight w = 0.8, the weight used for expansion is 0.7*0.8 = 0.56. In other words, the model is used to increase or decrease the weight of terms used for query expansion, according to the history of the user. The vector Q' is broadcasted to the semantic search engines distributed through the virtual enterprise, as shown in Figure 4.

5. ARCHITECTURE

Taking a glance to the KEEN architecture, as shown in Figure 5, the most important element is the indexing and retrieval knowledge engine, which is able to perform analysis and indexing of documents. The engine is not a traditional IR-based search engine, which analyzes simple keywords, but enhance this representation with taxonomy information of the domain of the enterprise.
In order to satisfy the needs for modularity and scalability of the Virtual enterprise, we devised a distributed architecture, where multiple engines operates independently (see Figure 4). In this way, it is possible to employ security policies at company levels. In other words, it is each single company that decides what information export to the other enterprise's companies.
The Ontology, i.e., the taxonomy references, and Lexical entities, i.e., the textual information extracted from documents in taxonomy's categories, represent the internal knowledge-base of the system. They cooperate in order to allow the *semantic mapper*, that is, the described knowledge indexing module, to analyze a document and assign it the meta-information used during the retrieval. Such meta-information will be stored by the Indexer inside the Annotation Repository.
Annotations are at the foundation of the knowledge used by the Semantic Search Engine, i.e., knowledge querying module, for obtaining information related to the user queries.

Finally, the Substructure Extractor and Retriever (SER) allows the system to autonomously search the Virtual Enterprise's network and the Web for documents related to the enterprise's domain.

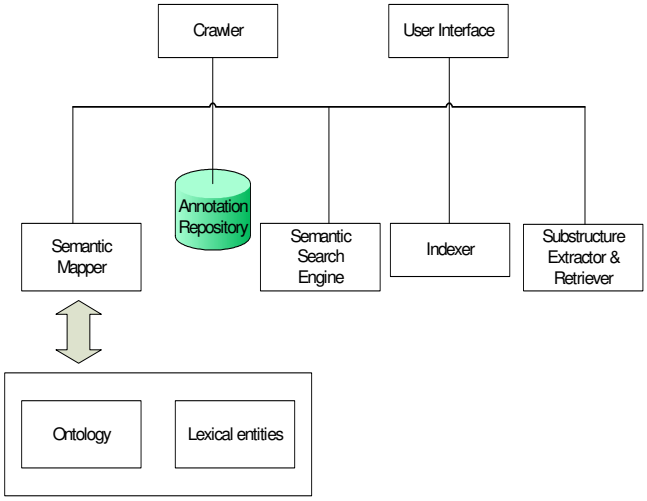

Figure 5: KEEN Knowledge Engine Architecture

After the SER has retrieved a set of documents, the Semantic Mapper extracts the most related categories from the internal taxonomy/ontology and stores this meta-information to the Annotation manager.
Each time a user of the Virtual enterprise submits a query, the semantic search engine retrieves all the meta-information stored in the annotation repository and the model of the user. The matching between the two components produces the personalized result list.
In this work, we have briefly described some of the modules, while others, e.g., Substructure Extractor & Retriever, have been ignored due to lack of space.

7. EVALUATION TEST BED

Due to the nature of the prototype, i.e., a distributed environment with different kinds of knowledge representations and typologies of users, we have decided to evaluate the system in a real scenario. In this way, we are able to evaluate all the available features through precision and accuracy measures, and the analysis of ad-hoc user questionnaires.

6. CONCLUSION

The goal of KEEN project is studying and devising methodologies and techniques for distributed management of knowledge in the context of a Virtual enterprise, favoring synergies and the interaction among different companies.
In particular, we have introduced a knowledge indexing and retrieval engine used for manage the information stored in a company. This engine is used by the users to retrieve documents related to the current needs.
The retrieval is enhanced with semantic metadata extracted by means of information extraction techniques. Moreover, the system employs a user modeling component to adapt the human-computer interaction during information seeking activities.

7. ACKNOWLEDGEMENT

We would like to thank all partners involved in the KEEN project:

- Camporlecchio Educational S.r.l.
- Sviluppo Italia Calabria Business Innovation Centre S.c.p.A.
- Mo.M.A S.r.l. – Modelli Matematici e Applicazioni
- Italdata S.p.A.
- Siemens IT Solutions and Services S.p.A.

8. REFERENCES

[1] M. Alavi and D.E. Leidner, "Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues". **MIS Quarterly**, Vol. 25 No. 1, pp. 107-136, 2001.

[2] F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi and P.F. Patel-Schneider (Eds.), **The Description Logic Handbook: Theory, Implementation, and Applications**. Cambridge University Press, 2003.

[3] P. Brössler "Knowledge Management at a Software Engineering Company - An Experience Report", **Workshop on Learning Software Organizations**, LSO'99, Kaiserslautern, Germany, pp. 163-170, 1999.

[4] P. Brusilovsky and M. Maybury, "From adaptive hypermedia to the adaptive web". **Communications of the ACM**, Vol. 45 No. 5, pp.30–33, 2002.

[5] F. Y. Y. Choi, "Advances in domain independent linear text segmentation", In **Proc. of the 1st conference on North American chapter of the Association for Computational Linguistics. Seattle, Washington**, pp26–33, 2000.

[6] C. W. Choo, "Information Management for the Intelligent Organization: Roles and Implications for the Information professions". In **Proc. of the 1995 Digital Libraries Conference**, Singapore, March 1995.

[7] T.H. Davenport and L. Prusak, **Working Knowledge: How Organizations Manage What They Know**, Harvard Business School Press, 1997.

[8] P. Ferragina and A. Gulli, A personalized search engine based on web-snippet hierarchical clustering. In **Proc. Of the 14th World Wide Web Conference**, 2005.

[9] G.W. Furnas, T.K. Landauer, L.M. Gomez, and S.T. Dumais, "The vocabulary problem in human-system communication". **Commun. ACM**, Vol. 30 No. 11, pp. 964–971, 1987

[10] B.R. Gaines, M.A. Musen and R. Uthurusamy, "Artificial Intelligence in Knowledge Management", **Technical Report SS-97-01**, Stanford University, 1997.

[11] S. Gauch, M. Speretta, A. Chandramouli and A. Micarelli, "User Profiles for Personalized Information Access". In Brusilovsky, P., Kobsa, A., Nejdl, W., eds.: **The Adaptive Web: Methods and Strategies of Web Personalization**. Volume 4321 of Lecture Notes in Computer Science, pp.54-89. Springer-Verlag, Berlin Heidelberg New York (2007).

[12] A. Gulli and A. Signorini, "The indexable web is more than 11.5 billion pages". In the **Proc. of the 14th international conference on World Wide Web (WWW05)**, pp902–903, New York, USA, 2005.

[13] R. Hartley, R. Kelsey, and R. Webster, "Consolidating Multi-Source and Multi-Media

Knowledge". In **AAAI Spring Symposium Articial Intelligence in Knowledge Management**. AAAI, March 1997.

[14] N. R. Jennings and M. Wooldridge, "Software Agents". **IEE Review.** Vol. 42 No. 1, pp17-21. January 1996.

[15] T. Joachims, "Text categorization with support vector machines: learning with many relevant features". In **Proc. of ECML-98, 10th European Conference on Machine Learning**, pp.137–142, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE. Published in the "Lecture Notes in Computer Science" series, number 1398. 1998.

[16] T. Joachims, "Transductive inference for text classification using support vector machines". In **Proc. of ICML-99, 16th International Conference on Machine Learning**, pp.200–209, Bled, SL, 1999. Morgan Kaufmann Publishers, SF, US, 1999.

[17] C. Johansson, C. and P. C. M. Hall, "Talk to Paula and Peter - They are Experienced"", **Workshop on Learning Software Organizations**, LSO'99, pp. 171-185, 1999.

[18] D. Kelly and J. Teevan, "Implicit feedback for inferring user preference: a bibliography"**. SIGIR Forum,** Vol. 37 No. 2, pp.18–28, 2003.

[19] G. Lawton, "Knowledge Management: Ready for Prime Time?" **IEEE Computer**, Vol. 34, No. 2, pp. 12-14, 2001.

[20] C. Moore, "Diving into Data". **InfoWorld**. October 25, 2002.

[21] I. Nonaka and H. Takeuchi, **The Knowledge-Creating Company**, Oxford Univers83(e)5.38597(r)-23.374.11624(24 -9.96 Td [(u)-20.8545(r)-9.66764n)-20.8545(n)-20.8545(4)-20.8545(,)3.27944

1902.

[21]