Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 19 (2013) 977 – 983

Procedia Computer Science

The 3rd International Symposium on Frontiers in Ambient and Mobile Systems

(FAMS 2013)

A new digital conceptual model oriented corporate memory constructing: Taking Data Mining models as a case

Choukri Djellali[a,b]

[a]*LATECE UQAM, 201, PK 4470, Président Kennedy Montréal (Québec) H2X 3Y7, Canada*
[b]*LANCI UQAM, C.P. 8888 Succ. Centre-ville Montréal (Québec) H3C 3P8, Canada*

Abstract

The integration of knowledge can be considered as a guideline for managing problems that occur in the task of knowledge management, and more particularly, in the collaborative decision-making. Integration is necessary because it allows communication between different sources. Most of the proposed approaches provide limited support for all activities of the engineering process, in particular, the phase of integration. We propose a new approach to treat the integration of the corporate knowledge. This model exploits indexation techniques and natural language processing to increase productivity of knowledge engineering task during the integration of conceptual model. Our integration system offers several advantages, these include speed search due to the structure and integrity of indexing.

Selection and peer-review under responsibility of Elhadi M. Shakshuki

Keywords: preprocessing, integration, corporate memory, data mining, ontology, information retrieval, machine learning.

1. Introduction

In recent years, knowledge integration has become a critical success factor for companies. The increasing complexity of products, globalization, Web crawling, social networks, virtual organizations, electronic publishing, customer orientation, digital enterprises and the explosion of the Web/IntraWeb requires a comprehensive and systematic integration of knowledge within a company. The solutions are often built around a corporate knowledge integration system [11], [18]. This model includes semi-formal or formal mechanisms to facilitate retrieval, broadcasting and reuse of knowledge by members of the company to solve their tasks.

The paper is organized as follows: In Section 2, we present our research questions and the problematic of the knowledge integration. The current state of the art in knowledge integration is given in Section 3. The conceptual architecture of the integration system is given in Section 4. Before we conclude, we give

Selection and peer-review under responsibility of Elhadi M. Shakshuki
doi:10.1016/j.procs.2013.06.136

in Section 5 a short evaluation with a benchmarking model for the corporate knowledge integration. Finally, a conclusion (Section 6) ends the paper with future works.

2. Problem and research questions

The nature of the data present on the Web or IntraWeb is generally dynamic, physically distributed and heterogeneous (database, structured or not structured documents, bitmap, sound, video, etc.). To create and share information efficiently many problems must be solved.

Firstly, we must locate the source of information that could contain the data needed for a given task. Once the source of information has been found, the access to data must be provided. This means that each source of information found should collaborate with the retrieval system. The problem that may arise is heterogeneity: structural heterogeneity (heterogeneity schematic) and semantic heterogeneity (heterogeneity of data). The structural heterogeneity means that different information systems store their data in different structures. The semantic heterogeneity is related to the content of information and its intended meaning. This problem is known in the community of database systems such as interoperability problem [19], [4].

Secondly, dynamic environments such as IntraWebs require the improvement and evolution of the corporate knowledge to ensure that it reflects the needs analysis (the domain completeness) [22], [23].

3. The current state of the art in knowledge integration

Several studies of knowledge integration focused on the identification and integration of relevant knowledge to provide a richer understanding of a particular domain. Some of the most popular systems are SIMS, TSIMMIS, Garlic, ObjectGlobe [3], InfoMaster, DISCO, Tukwila, MIX, Clio, Xyleme [8], RETSINA [13], InfoSleuth [16], UMDL [1], KnowWeb [15] and FRODO [14].

It is relevant to consider the disadvantages that have been observed in the literature, in particular, the presentation (the implementation of data structure) and understanding of data (the semantic definition of data stored in the integration systems).

Most previous approaches provide limited support for all activities of the engineering process, in particular, the phase of integration. In these approaches, there are no built-in methods, or tools that combine different techniques and heterogeneous sources of knowledge with existing knowledge to accelerate the integration process.

4. The architecture of our integration system

One of the key considerations of our design is to find a method to acquire and store the content of a document. There are many textual APIs (Application Programming Interface) and we decided to use Lucene [5] for our project.

Figure (1) shows schematically the functional process of the system with use cases. The user sends a query string to the system interface for resource discovery. The use case (Analysis) deals with a query string and generates a retrieval expression. The retrieval module (Retrieval) searches the index file and submits documents to triage program. This module sorts the documents according to their relevance and submits resources to the interface. The system interface provides the documents to users based on preprogrammed formats. The administrator of the corporate memory (Administrator) manages a dynamic index that supports adding and retrieving documents from the index. He uses corporate memory management tools to inspect / unlock / optimize an index and also to boost, remove and restore documents. The visualization module (Visualization) presents to the user a list of documents substitutes to simplify the exploration. Thus, each document is represented by a short summary associated with the calculated similarity or probability of relevance for a particular query.

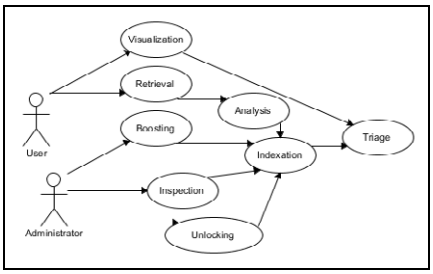

Fig. 1. The functional process of corporate memory

Figure (2) shows the architecture of the corporate memory composed of five modules: CRISP-DM-OWL ontology [(1)], acquisition, indexing, retrieval and interaction interface. The integration system uses the CRISP-DM-OWL ontology [12], [21] that provides a shared vocabulary for specifying the semantics. Therefore, the knowledge in the corporate memory is linked to a global ontology. This shared vocabulary describes the artefacts and the basic rules to improve the intelligence level of the system. It acts as a source of additional knowledge in the system. Communication between the components of the corporate memory is based on ontological artefacts. Before using the module provided by the Data Mining system, the user must first use the administration module to create an index of documents in the corpus. This is necessary to update the indexing model used in the retrieval module and the learning module of corporate ontology. When the index is created the system should maintain a temporal version to check if the document collection was modified after the creation of indexes. Once this step is completed, the user can use the Data Mining module to enrich the ontology. The descriptive clustering is used to generate new changes from documents available in the training corpus. All clusters are described by keywords (labels) representing their content. Labels and ontological artefacts are compared using an alignment process. The learning process uses the identified alignment rules to provide the necessary update. The reasoning techniques are used to check the terminology (T-BOX) / assertion (A-BOX) consistency of the updated ontology [10].

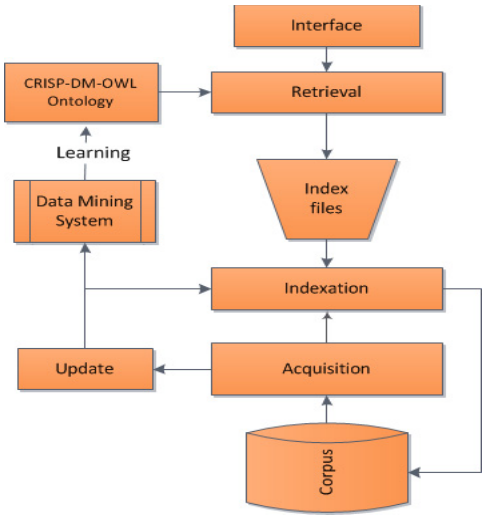

Fig. 2. Corporate memory architecture

(1) *http://www.elmanahel.ca/ontology/crisp-dm-owl.owl*

5. Experimentation and results

The integration process of corporate knowledge involves four main steps:

5.1. The text analysis

To ensure optimum use of integration techniques, pre-processing of data is essential for efficient data exploration. The negative dictionary and stemming are the most frequently used pretreatment techniques to remove noise.

- Negative dictionary: before creating the indexation of the document, it is necessary to delete all occurrences of these words. We used the Glasgow list [17] as a stop words list in our experiments. This list is widely used as English standard stop word; it covers a large number (351 stop words).
- Stemming: among several implementations of the stemming algorithms, we chose the version that was published by Martin Porter [6]. This version has the advantage of a clear separation between the substitution rules and procedures that test the conditions attached to a particular lexeme.

The training corpus consists of a set of IEEE abstracts divided in several categories. The number of documents in each category is highly unbalanced. The average length of the document in terms of words is 112.907 in the training set and 105.017 in the test set.
Thirty percent of the data are selected to test the model (no theoretical justification for this percentage). The data sample contains 94693 terms in the training set and 29676 terms in the test set. After pre-treatment of documents by tokenization, punctuation, negative dictionary and stemming, the bag of words contains 59240 terms representing the vocabulary in the training set and 19145 terms in the test set.
Figure (3) and figure (4) show the progress of the analysis through the documents in the training and test sets. The horizontal axis lists the documents in the training (test) set and the vertical axis indicates the accumulated vocabulary for each category {Punctuation}, {Stop word}, {Lexeme}, {Vocabulary} (respectively).

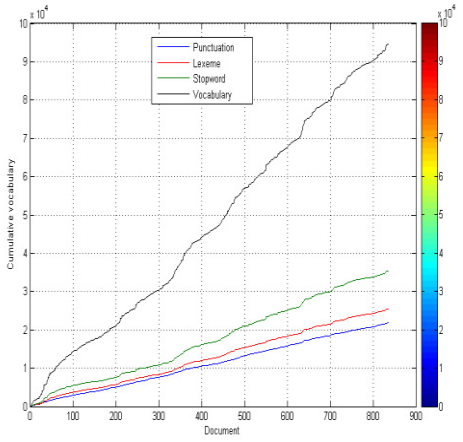

Fig. 3. The analysis of the training sample

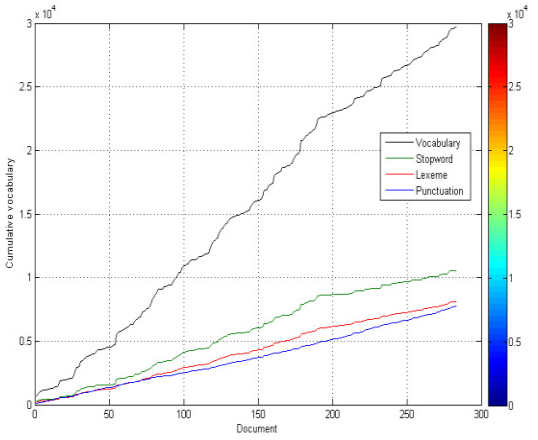

Fig. 4. The analysis of the test sample

5.2. Describe the relationship between the source text and the indexed text

In order to improve the index of the corporate memory resources, we analyzed two methods of indexing: compound and multi-file indexing [5].

Figure (5) and (6) show the indexing of documents in the training and test sets (respectively). The horizontal axis represents the document and the vertical axis shows the indexing time in milliseconds.
Indexing time of the training set is equal to 247 milliseconds with the compound method and 299 milliseconds using multi-file indexing method. To index the test set, the execution time is equal to 138 milliseconds during compound indexing method and 165 milliseconds during multi-file indexing method. In the light of the results it is undoubtedly to see that Multi-file indexing consumes more time because it stores many separated files by segment. Compared to the multi-file indexing, compound indexing reduced the number of index files because the compound structure encapsulates individual files into a single compound file format, thus it gives quick answers.

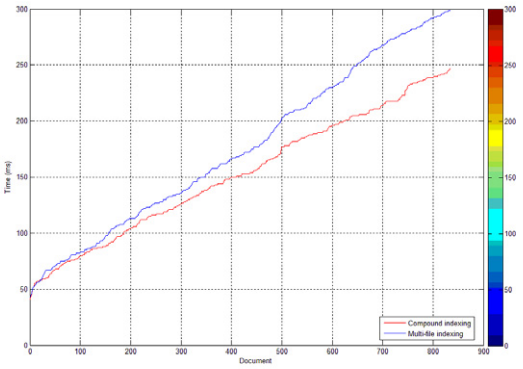

Fig. 5. Indexing time of the training set

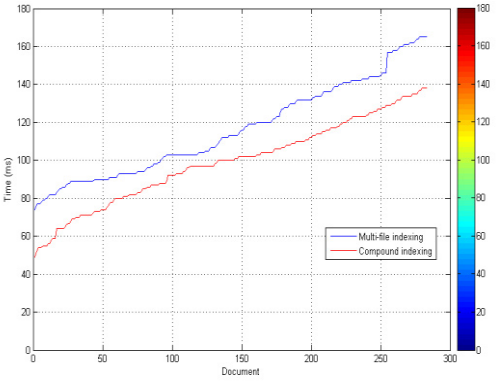

Fig. 6. Indexing time of the test set

Figure (7) and (8) show the allocation of memory (RAM) during compound and multi-file indexing. Due to the creation of new segments whenever documents are added to the index, there will be a variable number of files in memory with both methods.
To index the training corpus, the memory allocation for multi-file indexing is equal to 7438.28 kilobytes and 3447.53 kilobytes during compound indexing. For the test set, the memory allocation is equal to 1274.89 kilobytes during multi-file indexing and 1026.89 kilobytes during compound indexing. Compound indexing reduces the number of opened index files in memory because it contains a single composed file by segment. This optimized structure consumes less file descriptors and less computational resources during the indexing process. However, each segment of a multi-file indexing consists of several different files. Thereby, multi-file indexing does not improve performance and it is expensive.

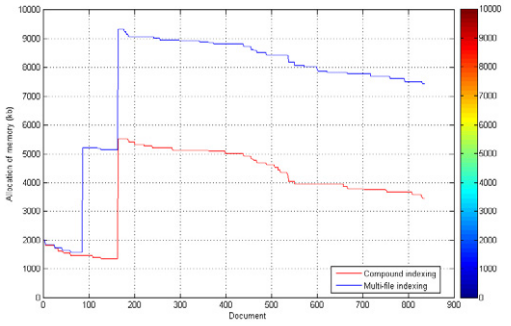

Fig. 7. The memory allocation of the training set

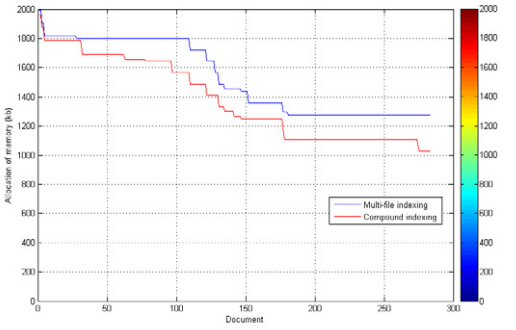

Fig. 8. The memory allocation of the test set

5.3. Registration of text in an index

The index structure is composed of five classes of data: directory, segment, document, field and term. The indexing module stores the entry in a compound structure. The optimal structure of the inverted index allows efficient use of disk space, search by keyword and quick answers. Each segment is an independent index containing a subset of indexed documents. Most index files are grouped into a compound file. Therefore, the performance of indexing and search are improved. With the hash function, each term generates a single value stored in the index. To optimize the structure of the index, the indexing module selects and merges segments. Selection of segments to be merged is governed by alignment policy and execution is performed by a fusion module. The administrator of the corporate memory uses GUI tools system (Luke, LIMO, Hadoop, Zipf, etc.) to inspect the details of the index from desktop or online applications [5].

5.4. Retrieval

The retrieval algorithm is based on the vector space model (VSM). The document and the query are represented by two vectors. Therefore, the similarity between a document and a query is calculated by the cosine of the angle between the corresponding vectors.
To evaluate the effectiveness of the corporate memory retrieval, we used measures of precision, recall and F-measure index that are widely used in information retrieval, natural language processing and machine learning [7], [20].

Table 1. The effectiveness of retrieval

Precision	Recall	F-measure
0.56	0.91	0.69

Experiments (Table 1) show that the integration system has good retrieval performance, which can provide a system of effective corporate knowledge retrieval.
One of the main advantages of our integration system is the construction of query analyzer for specifying and combining complex query strings. This feature was an ingredient key in the identification of documents in the corporate memory.
In addition, the retrieval module supports several types of advanced search, in particular, multiterm queries, phrase queries, wildcards, fuzzy queries. These search process are specified in several APIs, particularly, Phrase Query, Fuzzy Query, Prefix-Query, Range Query, Filtered Query, Boolean Query, WildcardQuery, etc [5].
Experiments show that the system has good indexing and good retrieval performance, which can provide a system of effective integration.
However, the recognition of corporate memory resources should be improved. The purpose of our next work is to propose a new method to find useful resources by exploiting the underlying content structure of the file to extract its text and metadata information.

6. Conclusion

Knowledge integration involves several challenges: heterogeneity of representation formalisms, languages and/or tools, lexical and semantic problems, implicit axioms in each system, loss of knowledge due to the interpretation and the evolution of knowledge.
We have designed and implemented an integration system that is based on the acquisition, indexing and retrieval. The integration system provides users with fast, flexible access to information through a query

unified interface. It allows users to specify the type of information required without providing details instructions on how to obtain this information. Indexing and retrieval are derived statistically by the co-occurrence of terms in documents and queries. Indexing involves assigning an index terms to unstructured text resources available in the corporate memory. These terms are then used to access the corporate resources using a table of indexes represented by an inverted file. Compound indexing method is used to reduce the number of opened index files in memory because it contains a single composed file by segment. This optimized structure consumes less file descriptors and less computational resources during the indexing process. However, each segment of a multi-file index consists of several different files. Thereby, multi-file indexing does not improve performance and it is expensive. Interoperability is considered the main application of CRISP-DM-OWL ontology particularly in the task of corporate knowledge integration.

The main advantages of our system integration are focused on the creation, maintenance and accessibility of the inverted index.

References

[1] N R Adam, V Atluri, and I Adiwijaya. SI in digital libraries. Communications of the ACM, 43(6):64-72, 2000.

[2] M Baena-Garcia, J M Carmona-Cejudo, G Castillo, and R Morales-Bueno. TF-SIDF: Term frequency, sketched inverse document frequency. In Intelligent Systems Design (ISDA),11th International Conference, pages 1044-1049, 2011.

[3] A Buccella, A Cechich, and N R Brisaboa. An ontology approach to data integration. Journal of computer science and technology, 3(2):62-68, 2003.

[4] Alfred Ka Yiu Wong, Pradeep Ray, N. Parameswaran, and John Strassner. Ontology Mapping for the Interoperability Problem in Network Management. IEEE Journal on Selected Areas in Communications Page(s): 2058-2068, 2005.

[5] E Hatcher, O Gospodnetic, and M McCandless. Lucene in action, 2004.

[6] B Issac and W J Jap. Implementing spam detection using Bayesian and Porter Stemmer keyword stripping approaches. In TENCON, 2009 IEEE Region 10 Conference, pages 1-5, 2009.

[7] William B Frakes, Ricardo A., and Baeza-Yates. Information Retrieval: Data Structures & Algorithms. Prentice-Hall, 2000.

[8] M L Lee, L H Yang, W Hsu, and X Yang. XClust: clustering XML schemas for effective integration. In Proceedings of the eleventh international conference on Information and knowledge management, pages292-299. ACM, 2002.

[9] G Salton, A Wong. A vector space model for automatic indexing. Communications of the ACM, 18(11):613-620, 1975.

[10] Djellali Choukri; Jean-Guy Meunier and Sylvain Delisle. A new approach to the evolution of Data Mining ontology, EGC-M12. The 3rd International Conference on the Extraction and Management of Knowledge - Maghreb, Tunisia, pages 100-107, egcm 2012. https://oraprdnt.uqtr.uquebec.ca/pls/public/gscw031?owa_no_site=21&owa_no_fiche=58.

[11] E K Mohamed, E F Abdelaziz, and C Ellis. Enterprise work□flow, corporate memory, and decision-making. In Multimedia Computing and Systems (ICMCS), 2011 International Conference on, pages 1-8, 2011.

[12] Yanfen Shen. A formal ontology for Data Mining : principles, design and evolution: thesis presented at University of Quebec at three rivers 2007.

[13] K Sycara and A S Pannu. The RETSINA multiagent system (video session): towards integrating planning, execution and information gathering. The second international conference on Autonomous agents,pages 350-351. ACM, 1998.

[14] L Van Elst, A Abecker, and H Maus. Exploiting user and process context for knowledge management systems. In Workshop on User-Modeling for Context Aware Applications, 8th International Conference on User Modeling, Citeseer, 2001.

[15] F Vernadat. Enterprise modelling and integration. In Proceedings of the IFIP TC5/WG5, volume 12, pages 25-33, 2002.

[16] H Wache, T Voegele, U Visser, H Stuckenschmidt, G Schuster, H Neumann, and S Hübner. Ontology-based integration of information-a survey of existing approaches. In IJCAI-01 workshop: ontologies and information sharing, volume 21, pages 108-117. Citeseer, 2001.

[17] A N K Zaman, P Matsakis, and C Brown. Evaluation of stop word lists in text retrieval using Latent Semantic Indexing. In Digital Information Management (ICDIM), 2011 Sixth International Conference on, pages 133-136, 2011.

[18] Li Zhang, Bao-wei Liu, and Ye-zhuang Tian. Empirical Study on Organizational Memory Constructive Factors. In Management Science and Engineering, 2007. ICMSE 2007. International Conference on, pages 1487-1492, 2007.

[19] Li Jingxia; Zhong Yun Design and implementation of the object-oriented EMS real-time database (CICED), 2010 China International Conference on Geosciences ; Nuclear Engineering ; Power, Energy, & Industry Applications.

[20] C TLu, M Shukla, SH Subramanya, and YWu. Performance evaluation of desktop search engines. pages 110-115. IEEE, 2007.

[21] T R Gruber. Toward principles for the design of ontologies used for knowledge sharing. International journal of human computer studies, 43(5):907-928, 1995.

[22] Li Xue; Cui DuWu; Chen Hao; YongQin Tao. Research of knowledge evolution system on simulation of human scientific knowledge (ICNC), Sixth International Conference on Natural Computation Communication, Networking & Broadcasting 2455-2461, Volume:5, 2010.

[23] A Maedche and S Staab. Ontology learning for the semantic web. Intelligent Systems, IEEE, 16(2):72-79, 2001.