

Integrating structured and unstructured data in a business-intelligence-system

Carsten Felden

Technische Universität Bergakademie Freiberg

Abstract

The recent liberalization of the German energy market has forced the energy industry to develop and install new information systems to support agents on the energy trading floors in their analytical tasks. Besides classical approaches of building a data warehouse giving insight into the time series to understand market and pricing mechanisms, it is crucial to provide a variety of external data from the web. Weather information as well as political news or market rumors are relevant to give the appropriate interpretation to the variables of a volatile energy market. Starting from a multidimensional data model and a collection of buy and sell transactions a data warehouse is built that gives analytical support to the agents. Following the idea of web farming we harvest the web, match the external information sources after a filtering and evaluation process to the data warehouse objects, and present this qualified information on a user interface where market values are correlated with those external sources over the time axis.

1. Introduction

This paper presents a new approach to enhancing data warehouses by adequate and highly-related information from Internet sources. An integrated process of searching and semi-automated evaluation will be shown to bring more precise information from outside into such a data warehouse. The largest German utility company (RWE AG, Essen, Germany) runs a market data information system to give the highest possible analytical support to their energy traders (Felden 2002). The kernel is a data warehouse which stores harmonized heterogeneous data derived from internal trading systems and external information brokers. The integrated data can be differentiated into structured data (e.g. accountings which are computer processable) and unstructured data (e.g. text documents which are not computer processable).

The name “unstructured data” is based on the circumstances that the structure of a text document is not clear to an information system because text semantics cannot be understood by machines. The conceptual framework and the data model of this Systematic Analysis and Research Tool (SMART) for energy trading will be introduced in Section 2. Until now there have only been a few published approaches which tackle this problem domain and give approved solutions for the coupling of internal and external data. This means the already structured data from inside the company and unstructured data from the Internet. The classical approach is published by Hackathorn (Hackathorn 1998). He suggests that external data should be qualified and classified by a person who works as an information editor within the company. This idea is implemented for example in the tools *LIXTO*, developed by the University of Vienna, or *Dys-MIS*, which is used at the University of Bonn. A second approach derives from the idea of business information collection as part of the strategic enterprise management (SAP SEM™) initiative (Meier & Mertens 2001: 149). This process of integrating external (business) information into the data warehouse is not automated either but needs an editor workbench. In Section 3 we will show several methods which enhance the idea of web farming. Firstly we build a set of metadata-based descriptors to classify external information, secondly we evaluate a classification algorithm to select potentially interesting information and thirdly we implement a graphical user interface which connects the information sources to the time series stored in the data warehouse. Based on this integration process we will use ontology-based user profiling to support the triggering of unstructured data and the identification of early indicators in text documents. Section 4 will summarize these findings and give ideas for further developments.

2. Conceptual framework

The collection, reduction and selection of relevant information can only occur on the basis of consistent company-wide data retention. Due to the heterogeneous legacy systems, a systematic merging of relevant databases is necessary. The data warehouse concept is an attempt to efficiently manage and collect relevant information derived from the vast amount of data.

2.1 From data warehouse to active data warehouse

Devlin was the inventor of the idea of data warehousing (Devlin, 1997). The basic idea was to have data storage available for a huge amount of data that would give support analyzing data. Inmon identified four characteristics of a data warehouse, which are represented in his formal definition: "... a data warehouse is a subject oriented, integrated, non-volatile and time variant collection of data in support of management's decisions." (Inmon 2002: 33) The structure of a data warehouse is totally different from the structure of operational databases. A data warehouse differs due to the objective of an operational database by the type of data entered and their supply. The core of a data warehouse is a database, in which data from different operational systems are historically saved in different levels of aggregation.

Due to the fact that, as a rule, analysts make complex queries and demand intuitive working with the database, a multidimensional data model seems appropriate. Each combination of dimension e.g. region, time, or customer, characterizes an analyst's query. The complexity of a multidimensional structure is the result of the amount and the type of dimensions. Dimensions can be seen as the highest reduction level of data (Codd 1994). Therefore, two types of dimension can be differentiated. On the one hand, all elements of a dimension are equal; this means they all have the same granularity. On the other hand, there is a hierarchical relationship between them (Bulos 1998: 251). One example is the time

dimension. This dimension is the result of hierarchical aggregation starting from day to month, to quarter, and to year.

The dimensions of the described multidimensional data model are the basis of the integration process of external information. If the retrieval query refers to dimension terms, the result must be linked to these dimensions. If the retrieval query results from individual dimension attributes, a coupling must be ensured to the explicit slice of the multidimensional data structure, in order to guarantee the context of the inserted information.

An active data warehouse is based on the idea of data warehousing. As described in the literature (Bishop 1995), an active data warehouse is a collection of data, which describes facts and business-relevant events. The user-specific behavior is implemented in the database, in addition to the classical approach of a data warehouse. It is a system in the sense of an active database which supervises potentially interesting events. If a defined situation occurs, the system will initiate appropriate actions. This behavior can be specified by database triggers in the form of an event-condition-action-rule (Elmasri & Navathe 2003). These triggers are database procedures which can be programmed by users directly. Nevertheless it should be stated that this activity is currently implemented only for structured data. Also unstructured data contain information of great significance for decision makers.

There is a need for personalizing in order to present unstructured data meeting managers' requirements. Such a form of individual adjustment is already well-known from the field of electronic commerce (Bulos 1998: 251). Personalizing is understood as the adjustment of services and information to the context of individual users or user groups. Each person has different roles in life; a rigid allocation to a single role does not reflect reality. User preferences must be collected and stored over time in order to select only that information which fits best. Interactive adjustments and variability of this framework have to be considered.

Language@Internet SV1-5/2006 (www.languageatinternet.de, urn:nbn:de:0009-7-3738, ISSN 1860-2029)

2.2 Information retrieval process

There already exist approaches to integrate structured and unstructured data in a data warehouse. This section presents an overview of the integration process details and functions which have to be realized. This has to be understood as a basis for the further development (e.g. Hackathorn 1998). The integration process is divided into four different steps. The dimensions of the implemented data model are the basis of the retrieval process of external data. In the first step, we use metadata of respective dimensions enhanced with attribute values to conduct a search with an Internet search engine (1). A web extracting and loading tool is applied in order to transfer information. If the coded retrieval query refers to the dimension itself, the result has to be linked with this dimension. If the coded query refers to the attributes of a dimension, the unstructured data have to be linked to the OLAP-slice to ensure the context of the collected information. All information is stored in the so-called data barn in a second step (2). The data barn stores the incoming Internet documents for further processing. The third step is the usage of a filter component (3), to classify whether the external data are appropriate for the data warehouse content or not. In the first implementation, a Multilayer Perceptron (MLP) with data descriptors and data quality parameters as input is used for the classification. Alternative classification algorithms like Support Vector Machines or Naïve Bayes (Sebastiani 2002) can be used as well. A recent research project has shown that the Naïve-Bayes algorithm achieves the best results in this context at present. This has to be validated regularly. Finally, the relevant data are stored in the multidimensional database (4). Both the relevant and the irrelevant data are necessary to improve the filter component. The generated rules are stored in the rule library. This automated integration process is illustrated in the following Figure 1.

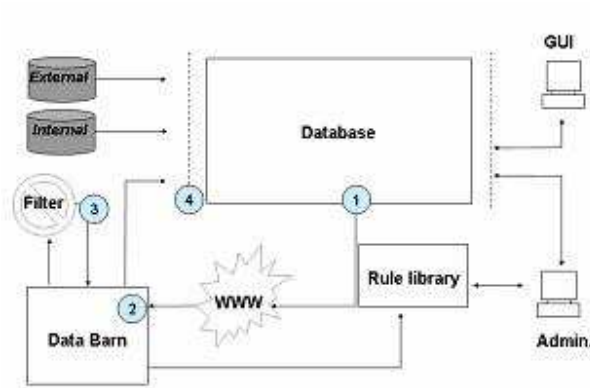


Figure 1: Integration process

3 Method

Internet documents have to be identified, classified and evaluated to clarify their grade of interest (Nayer 1993: 51). As the identification of information is carried by a meta-search engine, the following section concentrates on information evaluation. In principle the evaluation refers to text documents, since an automatic content-wise evaluation of picture, audio and video files is not feasible yet without difficulties. For this purpose, the construction of a rule generator is an essential problem. Rules have to be generated, which exclusively identify those relevant Internet data for decision makers in a filter.

3.1 Descriptor definition

In order to select a classification structure (this is called “information surrogate”), both an operationalization of the quality of information and methods of text mining must be taken into consideration. The final aim is to identify actual interesting pages and offer these to decision makers. The information surrogate consists of a vector capturing a subset of metadata, quality demands and conceptual extension. It is assumed that text information from the Internet is already available in a company and will be divided manually into interesting and uninteresting pages. The already existing pages are the training basis for identifying the descriptors that

characterize interesting and uninteresting texts. In order to determine the importance of the descriptors their frequency is calculated (Salton 1989; Salton & McGill 1983; Wooldridge, Müller & Tambe 1995).

Identical facts are presented differently on WWW-pages and accordingly the quality of information varies (Redman 2001). In order to determine the source type, the author of the website is taken into consideration. Therefore the type of organization and financing (e.g. advertising income) must be analyzed. The quality of the source is evaluated by correctness, actuality and navigation path.

In favor of the retrieval process, it is useful to complete the identified descriptors of texts with terms which improve their classification. The underlying problem is that qualified pages can be classified as interesting by means of the determined descriptors even though they are not really meaningful to the decision maker. This means that further characteristics have to be added manually.

3.2 Algorithms of the filtering process

The filtering process is described by generated rules that are stored in a library. The aim is to select the interesting information and then put it at the user's disposal. The task of text classification is to group documents into two disjoint classes: interesting and uninteresting documents. It is necessary to distinguish between the examination of documents and the setting up of classification criteria, both of which can be performed manually or automatically. Alternative automatic procedures are the Multilayer Perceptron (Bishop 1995), Rocchio algorithm, k-Next-Neighbour-Algorithm (kNN), Support Vector Machine (SVM), Naïve Bayes, Simple Logistics, Voted Perceptron, and HyperPipes which are already described in several publications (e. g. Sebastiani 2002; Freund & Schapire 1999; Hosmer & Lemeshow 2000; *Language@Internet* SV1-5/2006 (www.languageatinternet.de, urn:nbn:de:0009-7-3738, ISSN 1860-2029)

Pampel 2000; Rosenblatt 1958; Sheng 2005). These algorithms are chosen because studies of several other evaluations have shown that they are the ones examined most often (e. g. Collins 2002; Tveit 2003). Additionally, we are able to compare our results with other studies. All these approaches are based on a vector space representation of textual documents (Kobayashi & Aono 2004; Kamphusmann 2002; Colomb 2002).

There is not a single software suite with implementations of all algorithms, so that several software packages are required. The Java Tool Weka (version 3.4.2) implements the procedures kNN, Naïve Bayes and decision trees via J48, based on C4.5 (Witten & Frank 2000). SVM is provided by the C-program application SVMlight (Joachims 1998). The Rocchio algorithm is processed by a self-developed java application. The F_{β} -measure of van Rijsbergen is used for the evaluation of the classifications. It is calculated from recall and precision.

The majority of text classification studies use the Reuters textual data set which is highly normalized and standardized. Comparable pre-processing steps, as implemented in the context of the market data information system (MAIS), are insignificant in those evaluations. These aspects affect the result. Therefore the results of such studies are judged as insufficient.

The integration process is simulated in order to enhance the set of training data. The results of these queries are stored as text documents and define a set of training data. Finally, the documents are classified manually. The test data set was collected during a period of two months and covers 1,304 documents. This amount is small compared to other test collections. But it seems sufficient because only a binary classification is desired and the documents are much larger than those in the Reuters Corpus. Another reason is that the original problem domain (energy trading) needs just a small number of documents which have to be classified.

The necessary split into training, testing, and validating data is to be made in the relative ratio 50 : 20 : 30.

The data set developed contains more than 67,000 different words. Capitalized and non-capitalized words were accumulated to a single value. Variations of pre-processing steps to develop different input files for the tools were created. The variations are as follows: an implementation of a stop-word-list was developed in order to delete meaningless words. Further on, all hyphens between words have to be deleted and the remaining word parts are regarded as individuals. It is remarkable that during the conversion from HTML to TXT documents many special characters are produced automatically. The permissible terms are limited to a-z, A-Z, äüö, ÄÜÖ. Double 's' and 'ß' are treated equally. A '.' is determined as a decimal separator. A further step is the performance of stemming to improve the quality of the word list. Finally, the remaining words are integrated into a total list. Words which occur only once in a text document are deleted. Other words which only constitute the upper five percent in frequency per document are also deleted. Finally the word vectors are generated, which represent the input of the selected classification tools. The variations used obtain nine different input files with a number of terms between 10,343 and 33,776. Table 1 shows the final results of the text classification.

Test-No.	Term #	Algorithm	F_{β}
1	10,511	Naïve Bayes	0.7950
2	10,343	SVM	0.7868
3	15,676	Naïve Bayes	0.7810
4	31,602	Naïve Bayes	0.7870
5	33,247	Naïve Bayes	0.7870
6	33,392	SVM	0.7973
7	32,854	SVM	0.7844
8	33,602	Naïve Bayes	0.7865
9	33,776	Naïve Bayes	0.7865

Table 1: Classification results

Each algorithm was part of the evaluation, but only the text winners are shown in Table 1. Obviously, the Support Vector Machine of test No. 6 has the highest F_{β} -value. Most of the algorithms achieved their best result in this test (e. g. Voted Perceptron and Simple Logistic with a value of 0.7615, Rocchio with a value of 0.7261, or k-NN with 0.6395). HyperPipes (0.6865) and AdaBoost.M1 (0.7360) reached the best results in test No. 3. The MLP (0.5000) and J48 (0.7135) get their best results in test No. 1.

Looking at the classification including the pre-processing, the point of view has to be changed. Test No. 1 yields good results, but requires enormous additional pre-processing efforts. Looking at test No. 9, an F_{β} -value of 0.7865 can be found. This means comparable results can be gained with less pre-processing for the same algorithm. As a final result, the Naïve Bayes has the advantage of no mentionable pre-processing effort and a comparable classification result to Support Vector Machines. These results are comparable to other studies.

3.3 Visualization

A central problem consists in converting the data from heterogeneous sources into a single ultimately understandable user interface. Visualization needed in the market information system differs from the conventional concepts of generating inquiries and information retrieval by the ability of fast filtering, progressive change of search parameters and continuous change of goals and visual analysis of identified results. This leads to the following points, which must be implemented in the market information system: combining dynamic query filter, star field display and tight coupling (Ahlberg & Shneidermann 1999: 244; Nayer 1993: 51; Nielsen 1993). The display is illustrated in Figure 2.

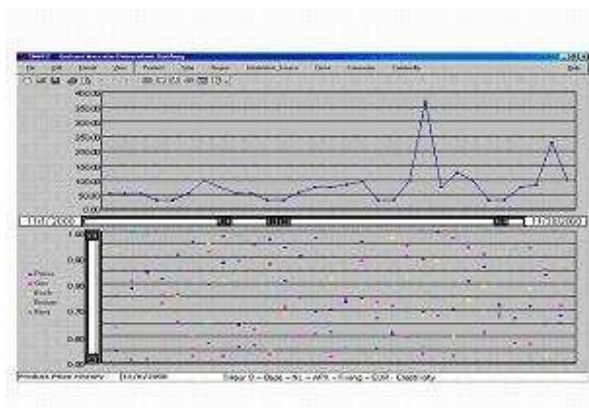


Figure 2: User interface with the star field display

The screen is split by a time slider in the center. The upper diagram represents the market data from well-known sources in a graph and the lower one represents the stored Internet documents in the form of a star field display (the y-axis reflects the measure of interest as a result of the classification process). Both diagrams are adjusted to the validity time of the data via time slider. The validity time of the Internet documents results from the date when the document was placed on the respective Internet server. The status bar on the bottom shows the selected dimension values. It is possible to change the upper and lower frame representations to

full size for further analysis. User studies have shown that early user integration by scenario and storyboard technique supports broad acceptance and intuitive usage of the system.

Of course, classification can be carried out by the filter component, but decision makers cannot get insight into all problems, because of time constraints. So, the individual information supply becomes focussed. This architecture is to be extended by two further components in order to fulfill the requirements of an active data warehouse. These are a user profile and an early-indicator profile.

3.4 Ontology-based user profiling

First of all, the user-oriented analysis of unstructured data requires a user profile and the knowledge of the valid problem-oriented variables of a specific problem domain. An analysis environment has to be developed to support managers' decisions not just short-term, but also in long-term (Sebastiani 2002: 1). Such an active data warehouse looks at specific situations and starts triggers to activate appropriate actions. This trigger component is already known for structured data. The basis is a rule knowledge, which is activated when a defined situation happens. Such a realization is called event-condition-action-rule (Elmasri & Navathe 2003). The usage of a trigger component for unstructured data is more complicated. It is already known that more than 80 percent of company information is only available as text documents (Simon 1960). The push principle demands sending appropriate information to specific users. The problem is that a trigger needs specified rules which are not available for text documents. Characteristic terms of documents can be compared, as a rule, with the information needs of a user and appropriate action can be taken.

The model of a user profile is based on the users' business role which is similar to a user-oriented filtering system. This model is built on a set of term vectors which can be used to

assign new text documents and generate recent and fitting information. The profile integrates descriptive information about the user and prescriptive information about the user preferences. The profiling can be divided into abstract and personal profiling (Abramowicz, Kalczyński & Węcel 2002).

Abstract profiling means that those terms are identified by documents of a set of training data which belong to a non-individualized class of people. This class represents e.g. the individual role of a person inside an organization. Personal profiling means the individual collection of terms which are rated as problem-characteristic by the decision maker. These terms can be arranged directly or determined automatically by selected documents. It has to be decided whether the structure of the profile is accomplished explicitly by the user or implicitly by observation of the user's behavior. It has to be considered that identical statements can be expressed by varying terms. This requires a procedure for analyzing text documents and determining varying expressions in text documents to create a thesaurus which should not just integrate descriptors but also phrases (Priebe & Pernul 2001: 73). There is a need to establish an interface between the documents and the users. On the one hand, the users should be able to create and maintain their profiles; on the other hand, it should be possible to validate the term vectors of the documents with the profile vectors. This is the reason for the usage of an ontology, which contains concepts to be a key element between the user and the information system. These concepts are e.g. terms and their relations and synonyms. An ontology is a designated key element, because it is understandable by both the user and the computer system.

If specific knowledge, for example from a division, is deposited with a certain terminology in an encyclopedia, the term "ontology" is to be used as description. This is specified by the differentiation into *Lightweight Ontology*, which illustrates a taxonomy, and

Heavyweight Ontology, which additionally integrates rules between concepts. The Greek-based term is not finally defined. However the following statement can be fundamentally met: an ontology is a formal and explicit specification of a hierarchical conceptualization. *Conceptualization* is to be understood as an abstract model of a part of the reality. *Explicit* means that the used concepts are defined. The attribute *formally* refers to the fact that these concepts must be machine-processable (Fensel 2004: 5). Domain-specific knowledge in the form of an arranged vocabulary is developed by ontologies (Kashyap & Sheth 2000: 11; Jain, Chen & Ichalkaranje 2002: 88). Linguistic styles are set in relationship to each other as objects in the form of concepts (Kamphusmann 2002: 32). A concept thus describes a certain object from a company-specific point of view (Kashyap & Sheth 2000: 11). The following table summarizes different ontology specifications. The specifications reflect the role of an ontology in an information system development process (Mizoguchi, Vanwelkenhuysen & Ikeda 1995: 46; van Heijst, Schreiber & Wielinga 1997: 183).

Ontology type	Ontology content
Knowledge Representation Ontologies	They are conceivable in different operational areas and offer only representation entities, without meeting a statement about contents.
Generic Ontologies	Such ontologies represent general terms and are even a basis for further ontologies of different domains.
Top-level Ontologies	These are superordinate concepts, which consist of a hierarchy of concepts.
Domain Ontologies	These work with terms and their relations within a defined domain, for example, the domain of medicine.
Task Ontologies	They define an activity and/or terms of

	an activity.
Method Ontologies	Their article is specific to the definition for problem solution methods so mentioned.
Application Ontologies	They combine domain and task ontologies and offer a specified system of concepts for an application.

Table 2: Ontology types (Fensel 2004: 5)

Ontologies are used, among other things, in order to describe the contents of text documents uniformly with concepts. If database queries are likewise transformed into concepts, this extends the resultant quantity to documents, which do not compellingly contain the special term of an inquiry (Abramowicz, Kalczyński & Węcel 2002: 86). Thereby the problem of synonymously-used descriptors and the quality of the resultant quantity can be solved. An ontology can improve communication between humans and machines, because an ontology contains machine-processable, semantic knowledge (Fensel 1998: 11). The concepts of an ontology form an abstract model. This model however is only valid for a defined range, because it is not possible to model all existing relevant relations between the integrated concepts. Due to this reason, high-quality results are only obtained if text documents belong to a specified range of topics for which the ontology was generated. The concepts contained in an ontology are connected by relations (Arjona, Corchuelo & Toro 2003: 62; Kashyap & Sheth 2000: 12). A company-specific taxonomy results from the structuring of the concepts (Feldman 2002: 754). It is important to find a suitable visualization of the ontology to support the selection of concepts in a certain technical area. Although a provided ontology can also be used for the extension of database queries. The terms of a query are assigned to a concept and

extended by synonyms of this concept and possibly subordinated concepts (ontology based query expansion), whereby the recall of the query can be improved (Kunze 2001: 392).

The concepts consist of a concept designation, a short description of the concept (Abramowicz, Kalczyński & Węcel 2002: 132), the concept-assigned set of synonymously-used terms and a set of directly subordinated concepts. The provided concepts can be divided manually into synonyms and subordinated concepts. In addition, a designation and a brief description of the concepts are made. Apart from the hierarchical relations between concepts, the influence of certain concepts on other concepts can also be illustrated (Pyle 2003: 183). The concepts can be assigned roughly to the company or the environment of an enterprise. In particular, the influences of environmental-referred concepts on enterprise-referred concepts are important and can be expressed by a so-called “influence relationship”. So for instance the environmental-referred concept *currency exchange rate* can be connected with the enterprise-referred concept *market price*. It has to be stated that a large number of concepts makes a clear grouping of concepts impossible. An illustration of all subordinated concepts with their various relations to other concepts at the same time can not easily be integrated in a model. Problems occur during the selection of the important concepts and relations for the ontology development as well as the extensive formulation expenditure. Hierarchically sorted listing of the individual concepts in different selection windows can be done as an alternative. However this graphic structure was re-used here, as the following figure shows.

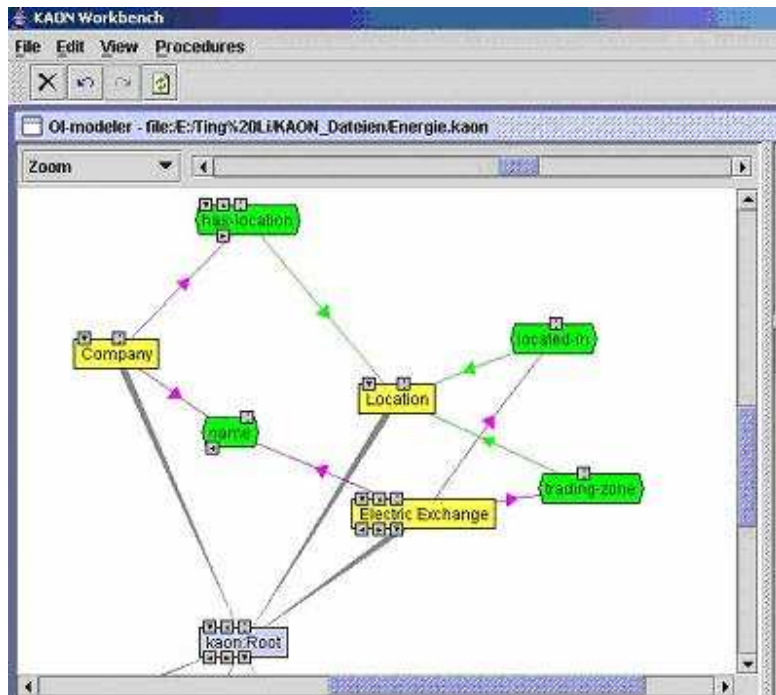


Figure 3: Ontology concepts and their relation

In context of the research, the figure shows part of an energy ontology with concepts, relations and attributes as a visualization graph. Domain concepts (rectangles) and rank concepts (hexagons) are differentiated. The links between concepts is visualized by arrows. A further difference between a semantic net and this visualization form is that domain concepts display their attributes. The following figure exemplifies the lexical entry for the concept *location* of the energy ontology.

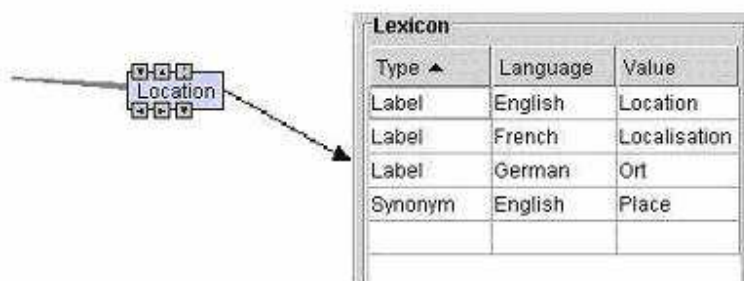


Figure 4: Lexical ontology entry

This form of the ontology modeling makes it possible to implement different techniques for analyzing data. This concerns a synonym search which scans text documents in different languages. For example, *Elektrizität* and *electricity* can be examined in such a way together. This is also similar for the analysis within a linguistic area, where for example, the concepts *electricity* and *power* can be regarded together. The possibility exists, additionally, of specifying important terms in some short sentences in the sense of a definition in order to increase the understanding inside the application. The dependencies of the elements of such a classification approach are shown in Figure 5.

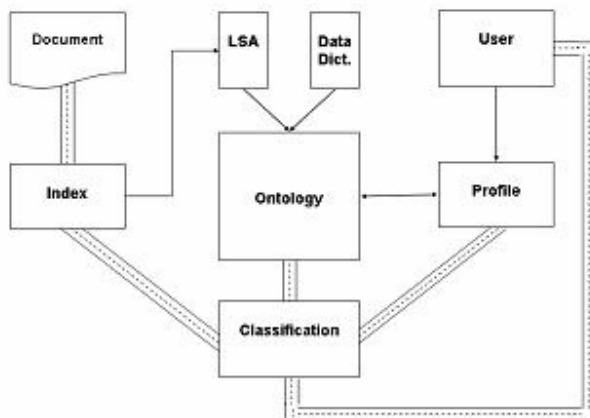


Figure 5: Ontology-based classification

The dotted lines represent permanently active relations and the arrows symbolize unique activities. As shown, a term vector is created, when a new document is stored in the data warehouse. This index has to be mapped with the user profile in order to classify the document by the usage of the ontology. The classification quality can be improved by the integration of company-specific knowledge which leads to the data dictionary, glossary, and ontology combining to form a company-wide unified system of concepts.

3.5 Triggering unstructured data

The user profile contains subject concepts and rule concepts. Both are based on the ontology, but they are examined in different ways. A trigger works for each profile. After the analysis of a text document (event), the trigger examines the stored thresholds. If the threshold is transcended (condition), appropriate information is sent to the user (action).

First of all, subject concepts store concepts with their term vectors. An event means that a new document has been inserted into the database and a document vector with descriptors generated. Document vector and profile vector have to be examined for condition fulfillment which is done by simple keyword matching. We measure the similarity between retrieved documents and profiles by the degree of correspondence between document indices and profile vectors. We use the cosine measure (standardized scalar product) to compute the difference between profile vector and document vector (Maedche, Pekar & Staab 2003: 301). We define that a condition is valid if and only if the lower boundary $<$ cosine measure assuming that the upper and lower boundaries are part of an extended profile. The similarity reaches the highest level if two vectors have the same direction. We can enhance the approach by adding relative frequencies of the terms in the document vector and/or personal weighting of the terms in the profile vector. In accordance to the user feedback, an appropriate update on the user profile and also on the training data has to be carried out.

We have to consider the three following sentences for the examination of the rule concepts which concepts are necessary to find word combinations. The idea is that in short sentences parts of the concept are sometimes in the following sentence. The sequence of terms is not important. It just has to be examined whether the concepts are existent or not. Finally, it has to be ascertained whether the user should be informed by a specified mail. The difference

to the cosine measure of the subject concepts is that a rule is fulfilled completely or not at all; we only have the values 0 or 1, not the interval.

Thus, an active data warehouse with structured and unstructured data can be realized and the trigger can be optimized by user feedback as evaluation.

3.6 Identification of early indicators

Structured and unstructured data of a data warehouse represent business activities in time. The identification of early indicators moves the focus into the future. Processes in enterprises have to be understood as a system of interacting elements which influence the planning (Heller 1992). This requires suitable techniques and tools such as simulation models in order to implement the complex tasks of planning and otherwise text mining to be able to identify text semantics.

It has to be stated that the usage of a simulation model is limited. It is not possible to develop a universally-valid and company-wide simulation model. The dynamic of the environment is too intensive and the number of parameters is too numerous. It has to be considered that the model has to meet the requirements of the model dynamics as well as those of the environment dynamics. This has a corresponding maintenance effort. To reduce such an effort, a feedback diagram for a specific model area is built where we have to identify the appropriate model elements, their connections and positive and negative feedback loops. These loops are analyzed qualitatively to determine dominating feedback loops and control loops.

The identification of effects in unstructured data presupposes that text semantics are recognized automatically. Additionally the quantification must be ascertainable to the respective effect in order to implement a simulation. Figure 6 shows the functionality for the identification of early indicators.

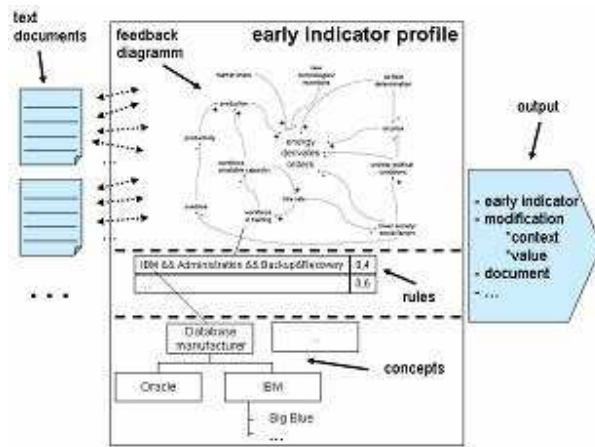


Figure 6: Functionality early indicator profile

The early indicator profile contains disjoint feedback diagrams for a specific subject. Each of the integrated parameters has its own rule library based on the developed ontology. The library also includes the influence value between parameters. If a new text document is stored in the data warehouse, the created term index is used to verify whether a model variable has to be modified or not. This happens according to the idea of the rule examination directly in the text document. The created output contains the identified early indicator, the accomplished modification, and the document itself.

4. Conclusion

The recent development of analytical information systems shows that the necessary integration of structured and unstructured data sources in data warehousing is possible. The usage of the market information system shows that the database improves the analytical power of decision makers, in order to recognize tendencies in the energy market promptly. Nevertheless the respective model and the system must grant high flexibility to adjust them to changing conditions in the energy market. Furthermore the activities on the energy market and the work of the analysts will enhance the system. Market information systems have to be optimized by

better evaluation of external information and automatization of process integration. Only documents of decision relevance should be delivered to the management. The ROI of data warehouse projects can be increased if event-based and accepted information improves the decision quality significantly. The information flow alignment in MAIS is equivalent to a classification problem. We assure this by using role profiles and embedded recommendation systems with a document trigger mechanism. Furthermore the use of a simulation method is tightly linked to this process by matching simulation variables to trigger conditions. The integration of metadata from a data warehouse, personalized search patterns and simulation variables give a powerful repository for active data warehousing. The theoretical approach and the benefit of creating interfaces for the meta models are part of further research. Nevertheless, decision makers gain individualized decision support and early insight into future developments.

The quality of classification algorithms must be examined in regular time intervals to guarantee best results. Therefore it is necessary to optimize the structure of the test environment which has to support intersubjective and intertemporal comparability of the test results. Classification evaluations are often accomplished; however these results are only important in the context of the selected data set and evaluation environment. In order to acquire concrete statements for MAIS, such an evaluation environment and the results are described in this paper. In order to find the perfect search terms, the most relevant documents are to be found so that not just the classification itself has to be optimized, but the Internet retrieval as well.

References

- Abramowicz, Witold, Pawel Jan Kalczyński & Krzysztof Węcel (2002). *Filtering the Web to Feed Data Warehouses*. London, Berlin: Springer.
- Ahlberg, Christopher & Ben Shneidermann (1999). Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays. In Card, Stuart K., Jock Douglas Mackinlay & Ben Shneiderman (eds.) *Readings in Information Visualization – Using Vision to Think*. San Francisco: Morgan Kaufmann. 244-252.
- Arjona, José Luis, Rafael Corchuelo & Miguel Toro (2003). A Knowledge Extraction Process Specification for Today's Non-Semantic Web. In Liu, Jiming, Chunnian Liu, Matthias Klusch, Ning Zhong & Nick Cercone (eds.) *2003 IEEE/WIC International Conference on Web Intelligence (Proceedings, Halifax)*. Los Alamitos: INSTICC Press. 61-67.
- Bishop, Christopher M. (1995). *Neural Networks for Pattern Recognition*. Oxford: University Press.
- Bulos, Dan (1998). OLAP Database Design - A New Dimension. In Chamoni, Peter & Peter Gluchowski (eds.) *Analytische Informationssysteme – Data Warehouse, On-Line Analytical Processing, Data Mining*. Berlin: Springer. 251-261.
- Codd, Edgar Frank (1994). *OLAP On-Line Analytical Processing mit TM/1*. Whitepaper.
- Collins, Michael (2002). Ranking Algorithms for Named-Entity-Extraction: Boosting and the Voted- Perceptron. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*. Philadelphia: University of Pennsylvania. 489-496.
- Colomb, Robert M. (2002). *Information Retrieval – The Architecture of Cyberspace*. London: Springer.

- Devlin, Barry (1997). *Data Warehouse – from Architecture to Implementation*. Reading, MA: Addison Wesley.
- Elmasri, Ramez & Shamkant Navathe (2003). *Fundamentals of database systems*. 4th edition. München: Addison Wesley.
- Felden, Carsten (2002). *Konzept zum Aufbau eines Marktdateninformationssystems für den Energiehandel auf der Basis interner und externer Daten*. Wiesbaden: DUV.
- Feldman, Ronen (2002). Text Mining. In Klösgen, Willi & Jan Zytkow (eds.) *Handbook of Data Mining and Knowledge Discovery*. Oxford: Oxford University Press. 749-757.
- Fensel, Dieter (1998). *Ontologies. A Silver Bullet for Knowledge Management and Electronic Commerce*. Berlin: Springer.
- Fensel, Dieter (2004). *Ontologies. A Silver Bullet for Knowledge Management and Electronic Commerce*. 2nd edition. Berlin: Springer.
- Freund, Yoav & Robert Schapire (1999). *Large Margin Classification Using the Perceptron Algorithm*. Machine Learning 37. Dordrecht: Kluwer. 277-296.
- Hackathorn, Richard D. (1998). *Web Farming for the Data Warehouse*. San Francisco: Morgan Kaufmann.
- Heller, Frank (1992). *Decision-making and leadership*. Cambridge: Cambridge University Press.
- Hosmer, David W. & Stanley Lemeshow (2000). *Applied logistic regression*. 2nd edition. New York: Wiley.
- Inmon, William (2002). *Building the Data Warehouse*. 3rd edition. New York: Wiley.
- Jain, Lakshmi C., Zhengxin Chen & Nikhil Ichalkaranje (eds.) (2002). *Intelligent Agents and Their Applications*. Studies of Fuzziness and Soft Computing 98. Heidelberg: WIC-Center Report (IEEE/WIC/ACM).
- Language@Internet* SV1-5/2006 (www.languageatinternet.de, urn:nbn:de:0009-7-3738, ISSN 1860-2029)

- Joachims, Thorsten (1998). *Text Categorization with Support Vector Machines: Learning with Many Relevant Features*. Forschungsbericht des Lehrstuhls VIII (KI), Fachbereich Informatik, Universität Dortmund.
- Kamphusmann, Thomas (2002). *Text-Mining. Eine praktische Marktübersicht*. Düsseldorf: Symposium.
- Kashyap, Vipul & Amit Sheth (2000). *Information Brokering Across Heterogeneous Digital Data. A Metadata-based Approach*. Boston: Kluwer (no. 20).
- Kobayashi, Mei & Masaki Aono (2004). Vector Space Models for Search and Cluster Mining. In Berry, Michael (ed.) *Survey of Text Mining. Clustering, Classification, and Retrieval*. ACM, New York: Springer. 103-122.
- Kunze, Claudia (2001). Lexikalisch-semantische Wortnetze. In Carstensen, Kai-Uwe, Christian Ebert, Cornelia Endriss, Susanne Jekat, Ralf Klabunde & Hagen Langer (eds.) *Computerlinguistik und Sprachtechnologie. Eine Einführung*. Heidelberg: Springer. 386-393.
- Maedche, Alexander, Viktor Pekar & Steffen Staab (2003). Ontology Learning Part One – on discovering Taxonomie Relations from the Web. In Zhong, Ning, Jiming Liu & Yiyu Yao (eds.) *Web Intelligence*. Berlin, Heidelberg, New York: Springer. 301-319.
- Meier, Marco & Peter Mertens (2001). The Editorial Workbench – Handling the Information Supply Chain of External Internet Data for Strategic Decision Support. *Journal of Decision Systems* 10(2): 149-174.
- Mizoguchi, Riichiro, Johan Vanwelkenhuysen & Mitsuru Ikeda (1995). Task Ontologies for reuse of problem solving knowledge. In Mars, Nicolaas (eds.) *Towards Very Large Knowledge Bases: Knowledge Building and Knowledge Sharing (KBKS'95)*. Amsterdam: University of Twente, Enschede, The Netherlands.

- Nayer, Madhavan (1993). Achieving Information Integrity: A Strategic Imperative. *Information Systems Management* 10(2): 51-58.
- Nielsen, Jakob (1993). *Usability Engineering*. Boston: Morgan Kaufmann.
- Pampel, Fred C. (2000). *Logistic Regression. A primer*. Thousand Oaks: Sage.
- Priebe, Torsten & Günther Pernul (2001). *Metadaten-gestützter Data-Warehouse-Entwurf mit ADAPTEd UML*. Proceedings 5. WI 2001. Augsburg: Physica. 73-86.
- Pyle, Dorian (2003). *Business Modeling and Data Mining*. Amsterdam: Morgan Kaufmann.
- Redman, Thomas C. (2001). *Data Quality – the field guide*. Burlington: Digital Press.
- Rosenblatt, Frank (1958). The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychological Review* 65: 386-408. [Reprint in Anderson, James & Edward Rosenfeld (1998) (eds.) *Neurocomputing: Foundations of Research*. Cambridge, MA: MIT Press.]
- Salton, Gerard (1989). *Automatic Text Processing – The Transformation, Analysis, and Retrieval of Information by Computer*. Reading: Addison Wesley.
- Salton, Gerard & Michael J. McGill (1983). *Introduction to Modern Information Retrieval*. Hamburg: Addison Wesley.
- Sebastiani, Fabrizio (2002). Machine Learning in Automated Text Categorization. *ACM Computing Surveys* 34(1): 1-47.
- Sheng, Jia. (2005). *A Study of AdaBoost in 3 D Gesture Recognition*. Available <http://www.dgp.toronto.edu/~jsheng/doc/CSC2515/Report.pdf> [last access 3 February 2005].
- Simon, Herbert A. (1960). *The New Science of Management*. New York: MIT Press.
- Tveit, Amund (2003). *Empirical Comparison of Accuracy and Performance for the MIPSVM classifier with Existing Classifiers*. Available: *Language@Internet* SV1-5/2006 (www.languageatinternet.de, urn:nbn:de:0009-7-3738, ISSN 1860-2029)

<http://www.idi.ntnu.no/~amundt/publications/2003/MIPSVMLClassificationComparison.pdf>

[last access 2 February 2005].

van Heijst, Gertjan, A. Th Schreiber & Bob. J. Wielinga (1997). Using explicit ontologies in KBS development. *International Journal of Human-Computer Studies* 46(2/3): 183-292.

Witten, Ian. H. & Eibe Frank (2000). *Data Mining: Practical machine learning tools with Java implementations*. San Francisco: Morgan Kaufmann.

Wooldridge, Michael J., Jörg P. Müller & Milind Tambe (1995). *Intelligent Agents II – Agent Theories, Architectures, and Languages, IJCAI'95 Workshop (ATAL), Montréal, Canada, August 19 - 20, 1995, Proceedings*. Berlin: Springer.

Submitted: 10.10.2005

Review results sent out: 07.01.2006

Resubmitted: 28.01.2006

Accepted: 02.02.2006