Ontology-Guided Data Acquisition and Analysis

Using Ontologies for Advanced Statistical Analysis

Dominic Girardi, Michael Giretzlehner
Research Unit Medical Informatics
RISC Software GmbH
Hagenberg, Austria
firstname.lastname@risc.uni-linz.ac.at

Klaus Arthofer
Department for Process Management in Healthcare
University of Applied Sciences Upper Austria
Steyr, Austria
klaus.arthofer@fh-steyr.at

***Abstract*—We present an ontology-based, domain independent data acquisition and preparation system, which is able to process arbitrarily structured data. The system is fully generic and customizes itself automatically at runtime based on a user-defined ontology. So it can be instantiated for any domain of application by defining an ontology for this domain. Furthermore, semantic rules can be integrated into the ontology to check the data's semantic plausibility. We plan to integrate statistical and data mining algorithms that take advantage of the structural ontology information to allow the user to perform (semi) automatic explorative data analysis. In this paper, the system is described in detail and a motivation for ontology-guided data analysis is given.**

Keywords - ontology-based data acquisition; semantic data quality; ontology-supported data analysis.

I. Introduction

Data analysis is one of the last steps in the process of a data-based research project. Surely, it is the most important one; the one that extracts the information out of the data which has been collected before. Still, the quality of the outcome is limited by the quality of the preceding steps, which are: data acquisition, data validation and cleaning, and data preparation. While syntactical data validation is well established, checking for data semantics is widely neglected; despite the fact that the problem of bad semantic data quality is serious and well known. Semantic data quality matters the data validity concerning the meaning and no syntactical aspects [19]. Using data of insufficient semantic data quality has fatal consequences regarding the usability of reports – keyword garbage in, garbage out. Consequently, data analysis cannot be considered as an isolated working step, but has to be integrated into a process containing all other steps, mentioned above.

We present an ontology-based, generic, web-based data acquisition and preparation system, which is able to store and process data of arbitrary structure and is therefore applicable to any domain of application. By modelling the domain of application as an ontology, the domain expert prepares the system for data acquisition. The rest of the system, including web-based user interfaces for data acquisition and import interfaces for importing electronically stored data, are created automatically at runtime, based on the ontology information. Furthermore, the system allows the definition of semantic plausibility rules to ensure not just the syntactic correctness of the data but also the semantic one. The formalized domain knowledge, which exists in the form of an ontology, is going to be used to guide and configure statistical analysis algorithms on the data. This allows the domain experts – who are most likely no IT experts – to set up and run their own data acquisition system without the need for an IT or database expert.

Section 2 contains an overview over related research projects. In Section 3, we provide a motivation and the theoretical background for our generic data acquisition and semantic checking infrastructure and provide key numbers of already running data acquisition projects. In Section 4, we describe that kind of statistical analysis features we want to integrate into the system and how the ontology helps to improve the results of the analysis. Our conclusions can be found in Section 5.

II. Related Research

Ontologies are widely used for flexible data integration, where data from multiple heterogeneous sources is mapped into a central ontology. In their one page position paper Zavaliy and Nikolski [12] describe the basic concept of an ontology-based data acquisition system for electronic medical record data. They use a very simple ontology, which contains four concepts (*Person, Hospital, Diagnosis* and *Medication*). The *Web Ontology Language* (OWL) [20] is used for modelling their domain. They point out, that the main reason for using an ontology-based approach is the need for adaptive data structures. This work is closely related to our work. However, their paper contains no information on how the data can be entered into the system, neither is there information about the system architecture or semantic data checking. There is also no information given on how adaptable their ontology is and if those four main concepts can be replaced or not. Despite the fact that they follow a very similar basic idea, our system is more extensive and matured. Guo and Fang [13] describe an ontology-based data integration system. They use ontologies to cope with the semantic and structural heterogeneity of data from different source applications. This is closely related to one aspect of our system, namely the automatically created data import interfaces. They can be used to import data from heterogeneous data sources. While data import is only one aspect of our system, Guo and Fang [13] focus on this matter and provide sophisticated concept mapping algorithms to improve the integration process. In their proposal, Dung and Kameyama [14] describe an ontology-based health care information extraction system. They modelled the medical domain with an ontology and use the semantic information of this ontology to extract information from texts. A so called "*New Semantic Elements Learning Algorithm*" extends the ontology semi-automatically. Although their approach differs in some ways from ours (information extraction vs. data acquisition and analysis), both share a common idea: Information from heterogeneous systems is stored into a central ontology. Nevertheless, their ontology is very closely related to their field of application, while our system is fully domain independent.

Lin et al. [15] provide an ontology for data mining processes. The knowledge, which is stored in the ontology, helps the user to decide which data mining algorithm should be used for the task on hand. Although we use the ontology primarily for data definition, the integrated data mining algorithms need to know how to treat the current data. This information can be derived from our ontology. So, while Lin et al. [15] use an explicit ontology for guiding the data mining process, our ontology will guide the process more implicitly be providing meta-information about data types, etc. Zheng et al. [16] identified the a gap in data mining processes that arises between technicians, who perform data mining, and the domain experts, who's knowledge is needed to interpret and guide it. They use two ontologies to cope with this problem: a domain-ontology, where domain experts enter their knowledge, and a task-ontology for choosing the best data mining algorithms for the current problem. While the latter one is closely related to [15], the first one tries to enforce the connection between technicians and domain experts. Viewed in this light, they try to solve the same problem as we do in the exactly opposite way. While they want to support the technician with domain knowledge, we want to enable the domain expert to run data mining algorithms. Nimmagadda and Dreher [17] give a good example how ontologies are used in praxis. They modeled the complex domain of oil production using an ontology to support the data integration and mining process.

Despite the fact that the concept of building an application upon an ontology is widely used, we could not find any system that implements this concept as consequently as we do. In most cases the systems are closely connected to their field of application and the domain in the background is not arbitrarily exchangeable. Furthermore, we could not find any ontology-based systems, which cover all aspects from data acquisition (manual via automatically created web interface and electronically via generated interfaces) to data storage, semantic data checking and data analysis.

III. Ontology-Based Data Acquisition

A. Technical Background

Whenever data of a non-trivial structure is collected for statistical analysis, a professional data acquisition and storage system is needed. Due to the semantic dependency of the systems' data structures from the domain of application they are usually inflexible and hardly reusable for other domains.

To overcome this drawback and resolve the dependency, we developed a data acquisition and storage system, which is not based upon a domain-specific data model, but on a generic meta-data model. The meta-data model is able to store the actual data model the form of an ontology.

According to the definition of Chandrasekaran et al. [1] ”*Ontologies are content theories about the sorts of objects, properties of objects, and relations between objects that are possible in a specified domain of knowledge.*”. Gruber [2] provides a more general definition: ”*An ontology is a specification of a conceptualization.*”. Technical details on the meta-model can be found in [18].

In this way, the system's meta-data model remains constant and independent from the domain of application. The process of defining the domain-specific ontology in the generic meta-data model is called instantiation of the generic meta-model by a domain specific ontology. This instantiation is performed by a domain expert with support of the Ontology Editor (see Section III.B). Furthermore, the collected data itself is stored into this meta-model.

The user interfaces for data input, including input forms, overview tables, search and filter functionality is automatically created at runtime, based on the ontology definitions. Numerous configurable properties ensure the possibility to customize the automatically generated graphical user interface (GUI).

B. System Architecture

The main purpose of the project is to create a system, which allows non-IT experts to set up and maintain their own professional data acquisition system for, e.g., research, clinical studies or benchmarking purposes. Furthermore, the system is able to store domain specific knowledge in terms of rules, in order to check the semantic quality of the stored data. The main parts of the system are:

- The Ontology Editor: This software allows the user to instantiate the generic meta-model with the ontology of his field of application.
- The Web Surface: The automatically created web surface allows the data collectors to enter their data into the system.
- The Semantic Check Engine: This part of the system checks each data record against the rules, defined by the user, to ensure its syntactic and semantic integrity.

In the following sections, important parts are described in detail.

C. Ontology Editor

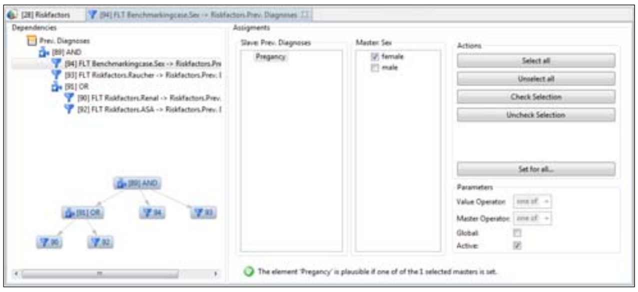
The Ontology Editor allows the definition and maintenance of the ontology. The domain-expert defines which data elements (classes) exist in his project, which attributes they contain, and how there are related to each other. Furthermore, the data type of each attribute has to be defined. The user can choose between text, integer, float, numerous date formats, and enumeration types. The latter ones are displayed as lookup tables. In order to keep large enumerations applicable they can be organized in hierarchical structures, resulting in taxonomies of enumerations. The ontology can be changed and adapted at any time.

Moreover, the Ontology Editor allows the display of the stored data sets and offers numerous filter, search and batch processing functions for administrating large data sets. Since the structure of the data depends on the actual ontology, all GUI structures (tables, headers, filter dialogs, etc.) are created at runtime, based on the ontology information.

D. Semantic Data Check

A benefit of ontology-based data-acquisition systems is the possibility to allow the domain-expert to manage enumeration types, without being dependent from an IT company to overtake this task. Due to the extensive use of adaptable enumerations the demand for free text input field is reduced to a minimum. Free text fields are the worst case for automatic processing for both: semantic checking and statistical analysis. Furthermore, the extensive use of enumerations forces the domain-expert to maintain these enumeration in order to keep them up to date and if the number increases – to organize them in meaningful hierarchies. This represents a central aspect of master data management.

For defining the rules for semantic data checks, the user has to establish so-called dependency relations (short: dependencies) among two processible attributes. Processible in this case means: not a free text attribute. One of the attributes is the master attribute; the other one the slave attribute. As the names suggest, the value of the master attributes defines the plausibility of the value of the slave attribute. In other words, the plausibility of the slave attributes depends on the value of the master attribute. E.g., the master attribute is the attribute *Gender* of the class *Patient*, and the slave attribute is the attribute *Diagnose* of the class *Disease*, then the diagnose *Pregnancy* is not plausible if the gender is *Male*. If one slave is controlled by more than one master, then these dependencies need to be connected by a logical operator AND or OR. This results in a logical tree of conditions, which is processed to determine the semantic plausibility of the given data set. Fig. 1 shows the configuration interface for this given example. The slave element (on the left hand side) is *Diagnose*, whereas the dependency is only set for the diagnose *Pregnancy*. The list on the right shows all possible master enumeration values for the master *Gender* (male and female). The state of the checkboxes indicate that a diagnose *Pregnancy* is plausible if the gender is *female* and not *male*. Asides from the actual configuration it shows the logical expression tree in which this dependency is embedded.


[28] Riskfactors
[94] FLT Benchmarkingcase.Sex -> Riskfactors.Prev. Diagnoses
Dependencies
Prev. Diagnoses
[89] AND
[94] FLT Benchmarkingcase.Sex -> Riskfactors.Pr
[93] FLT Riskfactors.Raucher -> Riskfactors.Prev. I
[91] OR
[90] FLT Riskfactors.Renal -> Riskfactors.Prev.
[92] FLT Riskfactors.ASA -> Riskfactors.Prev. I
[89] AND
[91] OR
94
93
90
92
Assigments
Slave: Prev. Diagnoses
Pregancy
Master: Sex
Actions
Select all
Unselect all
Check Selection
Uncheck Selection
Set for all...
Parameters

☑ female
□ male

Value Operator: one of
Master Operator: one of
Global: □
Active: ☑

The element 'Pregancy' is plausible if one of of the 1 selected masters is set.

Figure 1. Configuration dialog for a new dependency

One single attribute can be master of one dependency and at the same time slave of another dependency. So transitive dependency graphs can be created, which are processed from the leaves to the root of the tree using the constraint propagation scheme [3].

Before the data is used for statistical analysis, these checks are performed, and a detailed report is created. Check results are listed and the conflicts are described in detail. Combined with the check for syntactical correctness and the checks for static constraints (minimum and maximum values) the result of the semantic check provides a quality report of the current data set.

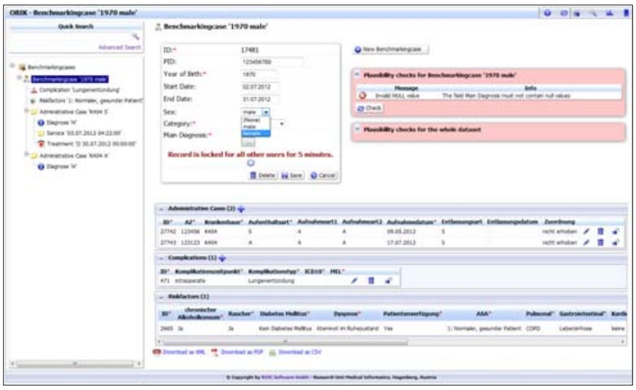
E. Experiences and Key Numbers

Currently, the described infrastructure is used to perform comparative benchmarking of surgical treatments of hospitals in Upper Austria. Each quarterly period the data is imported electronically from heterogeneous hospital information systems and supplemented with handwritten patient information by specially trained study nurses using the automatically generated web interface.

For each benchmarking cycle (3 months) about 1,500 medical cases are entered into the system, containing medical information as well as administrative data of a patient's treatment in hospital. An average case contains about 50 data elements (diagnoses, treatments, etc.), which results in about 75.000 data elements for a benchmarking cycle. Fig. 2 shows the automatically created web interface for this project. It displays one particular medical case. The tree on the left hand side visualises the whole case including all data elements, whereas its structure is derived from the ontology. The input form in the middle is also dynamically created based on ontology information and allows the editing of the currently opened case. The red fields on the right present the result of the semantic data check. While the colour indicates the result, a detailed list of errors is shown, when the user clicks on the field.

Before they are statistically analysed, semantic checks are performed. First runs of Semantic Check Engine emphasized the importance of semantic data checks and showed high error rates. Compares between manual and automated semantic checks showed a time saving of up to 15 minutes per medical case (about 15 minutes for manual checking by a domain expert vs. about 20 seconds for automated checking), resulting in an overall saving of more than three work weeks.

First clinic results are about to be published this year.


Quick Search
Advanced Search
Benchmarkingcases
Benchmarkingcase '1970 male'
Complication 'Lungenentzündung'
Riskfactors '1: Normaler, gesunder Patient'
Administrative Case '6404 S'
Diagnose 'H'
Service '03.07.2012 04:22:00'
Treatment 'D 30.07.2012 00:00:00'
Administrative Case '6404 A'
Diagnose 'H'
New Benchmarkingcase
Check
Delete
Save
Cancel
Download as XML
Download as PDF
Download as CSV

OBIK - Benchmarkingcase '1970 male'

Benchmarkingcase '1970 male'

ID:*	17481
PID:	123456789
Year of Birth:*	1970
Start Date:	02.07.2012
End Date:	31.07.2012
Sex:	male
Category:*	
Main Diagnose:*	

Record is locked for all other users for 5 minutes.

Plausibility checks for Benchmarkingcase '1970 male'

Message	Info
Invalid NULL value	The field Main Diagnose must not contain null values

Plausibility checks for the whole dataset

Administration Cases (2)

ID*	AZ*	Krankenhaus*	Aufenthaltsart*	Aufnahmeart1	Aufnahmeart2	Aufnahmedatum*	Entlassungsart	Entlassungsdatum	Zuordnung
27742	123456	6404	S	A	A	09.05.2012	S		nicht erhoben
27743	123123	6404	A	A	A	17.07.2012	S		nicht erhoben

Complications (1)

ID*	Komplikationszeitpunkt*	Komplikationstyp*	ICD10*	MEL*
471	intraoperativ	Lungenentzündung		

Riskfactors (1)

ID*	chronischer Alkoholkonsum*	Raucher*	Diabetes Mellitus*	Dyspnoe*	Patientenverfügung*	ASA*	Pulmonal*	Gastrointestinal*	Kardi
2665	Ja	Ja	Kein Diabetes Mellitus	Atemnot im Ruhezustand	Yes	1: Normaler, gesunder Patient	COPD	Leberzirrhose	keine

© Copyright by RISC Software GmbH - Research Unit Medical Informatics, Hagenberg, Austria

Figure 2. The automatically created Web Interface

IV. Ontology-Guided Data Analysis

The possibility to embed statistical analysis algorithms directly into the system offers numerous advantages for data analysts. We plan to extend our data acquisition system by a set of statistical analysis and data mining algorithms that enables analysts to get a quick overview of the data. The ontology information allows the (semi) automatic data mining, and helps to reduce the human bias on statistical analysis.

A. *Ad hoc Analysis*

In many cases, data acquisition and storage is done by different systems than data analysis. The data has to be extracted from the first and imported into the latter one. Although state-of-the-art data analysis systems like SAS, SPSS, etc., offer a big variety of data import interfaces, this process takes time. In the event that the most important statistical key features like descriptive statistic parameters, histograms, statistical hypothesis testing, etc., are directly integrated into the system, the analyst can answer simple questions right from within the system. The integrated filter and search functionalities help to extract the datasets of interest and automatically compute the most important key numbers. The data type of each attribute, which is provided by the ontology, helps to interpret the data and calculate the correct key features.

Moreover, correlations between all attributes and structural features (e.g., number of sub-elements of a certain class) can be computed automatically and all relevant results are presented to the user. In that way, unexpected correlations can be discovered, which reduces the human bias to statistical analysis.

B. *Semantic Data Quality*

As we motivated in Section III.D, semantic data checks help to increase the quality of statistical analysis. But there's a way to return this benefit in a way, that data analysis helps to improve data quality. Assume a strong linear correlation between two attributes a_1 and a_2. Consequently, records that show a configuration of a_1 and a_2 that sheers out of this correlation are suspicious and can be proposed for addition investigation.

Prototype tests on a data set of biometric measurements of over 1500 children showed very strong linear correlations between the different measurements of the human body. Several statistical outliners from these correlations were detected and revised. Although their actual data was within the defined ranges, these errors could be identified because their combination was implausible. Other tests showed that the weight of birth of children distributed normally – which we expected. More than ten input errors could be identified because of the significantly high distance of the data set from the mean of the distribution. With these two tests, we could show that even very simple statistical analysis can help to increase the quality of the stored data.

C. *Using Taxonomies for Higher Level Analysis*

The features described so far help to reduce the effort of data analysis by automation. For the following feature, the strong connection between the ontology and the analysis is essential.

Dealing with large enumerations quickly leads to sparse data sets. For examples, when dealing with medical data the diagnosis of a patient is often defined using the ICD-10 (International Classification of Diseases) catalogue, which consists of more than 12,000 entries. When, e.g., a Chi² [21] test is used to identify a correlation between the diagnoses and a treatment (coded with an enumeration type of a similar size), even with several thousands of data sets, the table would still contain lots of empty cells, and the result would be not be interpretable.

In this case, the hierarchically organized enumeration types help to reduce the number of rows and columns of the Chi² test setup. Since the hierarchical relations in the enumeration type is a IS-A relation the numerous enumeration values can automatically be summarized in meaningful groups; provided that the hierarchy was designed properly. So, instead of thousands of low level enumeration values less high level enumeration concepts are analysed; similar to the approach of Xiangdan et al. [4], where the concepts of ontologies are used to discover fewer high-level rules, instead of lots of low-level rules.

D. *Data Visualization including Structural Features*

One of main objectives of explorative data analysis is to get an overview of the data. A very popular instrument for this purpose is the self-organizing map (SOM), which was introduced by T. Kohonen [5]. A SOM maps an *n*-dimensional input space into an *m*-dimensional (usually *m*=2) output space, whereas it converts the nonlinear statistical relationships between high-dimensional data into simple geometric relationships [6]. Traditional SOMs work on numeric vectors for both input format and internal data representation. So, for using the SOM, the relational data has to be transferred into a numeric input vector. The ontology supports this transformation process in a way that allows persons, who are not necessarily IT experts, to perform this transformation just by choosing the attributes that shall be considered. Fig. 3 shows the result of the visualization of a medical dataset with a SOM.

However, the highly structured data has to be transferred into a one dimensional numeric vector, where most of the structural information is lost. In a further step, SOM algorithms for structured data (SOM-SD) will be evaluated

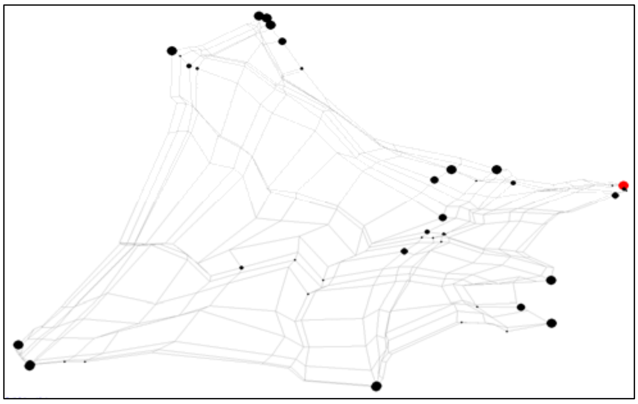


Figure 3. Data Visualization with a SOM

[7], [8], [9], [10]. For performing these algorithms access to the structural information is needed. So, this can only be automated and thus be made applicable for non-IT experts, by the strong linkage with the ontology.

V. CONCLUSION

In our research project, we could show that meta-model based systems help to reduce the setup effort of data acquisition and storage systems to a minimum. Additionally, the generic meta-model based approach allows the development of software features for classes of problem, not for just one single domain of applications. This guarantees a high degree of reusability of the system.

Furthermore, first semantic checks on real medical datasets showed high error rates concerning the sematic data quality in the tested hospitals, confirming what is conducted by Hüfner [11]. Without semantic data checks these error rates would influence the results of the analysis unnoticed, resulting in useless or suboptimal conclusions and consequences.

Statistical evaluation of the collected data is still performed in external systems, such as SAS, and SPSS, after exporting the data from the ontology based storage system. The planned statistical analysis features will never be able to replace these systems; neither is this our objective; but they provide an overview over the whole dataset before the analyst exports the data. Moreover, the strong linkage between statistical analysis and the ontology allows more sophisticated analysis with inclusion of structural and conceptional (enumeration type hierarchies) features. It also allows persons with limited IT skills to used highly complex algorithms like the SOM to explore their data.

REFERENCES

[1] B. Chandrasekaran, J. Josephson, and V. Benjamins. “What are ontologies, and why do we need them?” Intelligent Systems and their Applications, IEEE 1999, 14(1) pp. 20–26.

[2] T. R. Gruber. “A translation approach to portable ontology specifications”. Knowl. Acquis. 1993,5(2) pp. 199–220 .

[3] F. Rossi, P. van Beek, and T. Walsh. Handbook of Constraint programming. 1 edn. Elsevier, Amsterdam and Boston (2006)

[4] H. Xiangdan, G. Junhua, S. Xueqin, and Y. Weili. “Application of data mining in fault diagnosis based on ontology”, Proceedings of the Third International Conference on Information Technology and Applications (ICITA’05), 2005 IEEEes, Journal of Information Science, 2010 36:306

[5] T. Kohonen. “The self-organizing map”. Neurocomputing 21 1998, pp. 1-6

[6] T. Kohonen. Self-Organizing Maps – 3rd edn. Springer, 2001.

[7] M. Hagenbuchner, A. Sperduti, and A.C. Tsoi. “A self-organizing map for adaptive processing of structured data.” IEEE Transactions on Neural Networks 14(3) 2003, pp. 491-505

[8] M. Hagenbuchner and A.C. Tsoi. “A supervised training algorithm for self-organizingmaps for structures”. Pattern Recognition Letters 26(12) 2005 pp. 1874-1884

[9] M. Hagenbuchner, A. Sperduti, A.C. Tsoi, F. Trentini, F. Scarselli, and M. Gori. “Clustering xml documents using self-organizing maps for structures.” in: Workshop of the initiative for the evaluation of xml retrieval 2005, pp. 481 - 496

[10] M. Martin-Merino and A. Munoz. “Extending the SOM algorithm to non-euclidean distances via the kernel trick”. In Pal, N., Kasabov, N., Mudi, R., Pal, S., Parui, S., eds.: Neural Information Processing. Volume 3316 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg (2004), pp. 150-157

[11] J. Hüfner. „Datenqualität im Krankenhaus. Kostenvorteile durch ausgereifte Konzepte“, http://www.tiq-solutions.de/download/attachments/425996/Datenqualitaet-im-Krankenhaus_Jan-Huefner_08-2007.pdf (24.01.2012)

[12] T. Zavaliy and I. Nikolski. “Ontology-based information system for collecting electronic medical records data”. In: Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET), 2010 International Conference on, p. 125.

[13] G. Fangyua and Y. Fang. “An Ontology-Based Data Integration System with Dynamic Concept Mapping and Plug-In Management”. In: Information Technology, Computer Engineering and Management Sciences (ICM), 2011 International Conference on, vol. 3, pp. 324–328.

[14] D. Tran Quoc and W. Kameyama.” A Proposal of Ontology-based Health Care Information Extraction System: VnHIES.” In: Research, Innovation and Vision for the Future, 2007 IEEE International Conference on, pp. 1–7.

[15] L. Mao-Song, Z. Hui, and Y. Zhang-Guo. “An Ontology for Supporting Data Mining Process”. In: Computational Engineering in Systems Applications, IMACS Multiconference on, vol. 2, pp. 2074–2077.

[16] Z. Ling, L. Feng, and G. Hui. “The research of ontology-assisted data mining technology”. In: Information Management and Engineering (ICIME), 2010 The 2nd IEEE International Conference on, pp. 285–288.

[17] S. L. Nimmagadda and H. Dreher. „Petroleum Ontology: An effective data integration and mining methodology aiding exploration of commercial petroleum plays”. In: Industrial Informatics, 2008. INDIN 2008. 6th IEEE International Conference on, pp. 1289–1295.

[18] D. Girardi, J. Dirnberger and M. Giretzlehner. “Meta-model based knowledge discovery”. In: Data and Knowledge Engineering (ICDKE), 2011 International Conference on, pp. 8–12.

[19] A. Kurz. Data Warehousing. Enabling Technology. mitp-Verlag (1999)

[20] D. L. McGuinness and F van Harmelen. Owl web ontology language overview: W3c recommendation (10 February 2004).

[21] L. Fahrmeir, R. Künstler, I. Pigeot, and G. Tutz. Statistik – Der Weg zur Datenanalyse. 7th edn. Springer Heidelberg (2011), pp. 467–469