Towards Intelligent Data Analysis: The Metadata Challenge

Besim Bilalli[1], Alberto Abelló[1], Tomàs Aluja-Banet[1], and Robert Wrembel[2]
[1] *Universitat Politécnica de Catalunya, Barcelona, Spain*
[2] *Poznan University of Technology, Poznan, Poland*
{bbilalli,aabello}@essi.upc.edu, tomas.aluja@upc.edu, robert.wrembel@cs.put.poznan.pl

Keywords: Metadata, Data mining, Big Data analytics

Abstract: Once analyzed correctly, data can yield substantial benefits. The process of analyzing the data and transforming it into knowledge is known as Knowledge Discovery in Databases (KDD). The plethora and subtleties of algorithms in the different steps of KDD, render it challenging. An effective user support is of crucial importance, even more now, when the analysis is performed on Big Data. Metadata is the necessary component to drive the user support. In this paper we study the metadata required to provide user support on every stage of the KDD process. We show that intelligent systems addressing the problem of user assistance in KDD are incomplete in this regard. They do not use the whole potential of metadata to enable assistance during the whole process. We present a comprehensive classification of all the metadata required to provide user support. Furthermore, we present our implementation of a metadata repository for storing and managing this metadata and explain its benefits in a real Big Data analytics project.

1 INTRODUCTION

Our capability of gathering data has developed to the highest extents, whereas the ability to analyze it, lags far behind. Storing huge volumes of data is worth the effort only if we are able to transform data into knowledge. The process of transforming data into knowledge is known as Knowledge Discovery in Databases (KDD) and consists of the following steps: *data selection*, *data pre-processing*, *data mining* and *evaluation or interpretation (Fayyad et al., 1996).*

The need for knowledge discovery is rising tremendously. This is more noticeable nowadays thanks to the low-cost, distributed data storage and processing platforms (e.g., Hadoop). They allow storing and processing huge datasets on large clusters of commodity hardware. A Data Lake, for instance, is an important component of the data analytics pipeline in the world of Big Data. The idea is to have a single store of all the raw data (e.g., structured and unstructured) that anyone in an organization might need to analyze. However, the relevant data over which the analysis is going to be performed needs to be selected from the whole range of the available data. As the selection of data affects the results of the analysis, data needs to be thoroughly tracked in order to justify the results (e.g., lineage). The representation and the quality of data also affect the analysis. Raw data is often irrelevant, redundant, and incomplete and requires pre-processing. It is commonly known that 50-80% of data analysis time is spent on pre-processing. Once the data is pre-processed, there comes the difficult task of selecting the most adequate mining algorithm for a given problem. Many different algorithms are available and their performance can vary considerably. After data mining, the evaluation/interpretation step follows. The generated models need to be interpreted and/or evaluated to be understood by the user. All in all, the above mentioned steps indicate that KDD in general is an inherently challenging task. Therefore, users need to be thoroughly supported.

A lot of research has been done in this regard and systems that aim at providing user assistance have been developed. These systems are referred to as Intelligent Discovery Assistants (IDAs) (Bernstein et al., 2005). The driving factor for the user assistance is the metadata they consider. Yet, there is no agreement on which kinds of metadata need to be gathered and stored in order to provide user assistance. In this paper we tackle the problem by studying the types and roles of metadata. We observe that the meta knowledge considered in IDAs is not complete (e.g., domain knowledge and lineage is missing). Hence, we provide a classification of the metadata needed to support the whole process and discuss the implementation of our metadata repository.

Contributions. In particular, our main contributions are as follows.

- We identify and extend the metadata required for providing user support for the whole process of KDD including the very first step of data selection and we provide a classification of this metadata.
- We implement a metadata repository with the aim of storing and managing the metadata discovered and show its benefits in a real case scenario.

The rest of the paper is organized as follows. Section 2 discusses the related work. Section 3 presents an analysis of IDAs and briefly discusses the differences between different categories of these systems. Section 4 studies the metadata required for providing user support and shows examples of systems using the respective metadata. Section 5 contributes a classification of the metadata needed to support the whole process of KDD. Section 6 shortly presents the implementation of our metadata repository and its benefits in a real Big Data analytics project. Finally, Section 7 concludes the paper.

2 RELATED WORK

In (Foshay et al., 2007), a taxonomy of the end-user metadata with respect to data warehousing is given. This taxonomy is further extended in (Varga et al., 2014), where a metadata framework is provided to support the user assistance activities in the context of next generation BI systems. It provides a technical classification of the metadata artifacts required to enable user assistance in retrieving and exploring the data. The focus is on automating certain user related tasks with respect to queries (e.g., query recommendation). Whereas, we are studying and classifying metadata with the emphasis on how it can help the user during the different steps of KDD.

Another work that can be seen as closely related to us is (Serban et al., 2013). The authors provide a comprehensive survey of the systems that make extensive use of metadata to make the automation of knowledge discovery possible. The emphasis is put on explaining the architectures of the systems rather than on a comprehensive classification of metadata.

Finally, Common Warehouse Metamodel (CWM, 2003) provides the necessary abstractions to model generic representations of data mining models, however, the metadata considered does not cover the whole range of KDD steps. It is mainly focused on the metadata for the data mining step. Furthermore, the metadata is considered from the perspective of data interchange, which is how different systems can share and understand metadata with regard to data mining.

3 INTELLIGENT DISCOVERY ASSISTANTS

The KDD process is challenging for novice users. As already stated in Section 1, the most prominent works done in terms of providing helpful assistance to the users are through IDAs. In order to complete our study on the metadata needed for the user support we have to know how and to what extent this metadata is used by different IDAs. Depending on the core techniques and metadata used, IDAs can be divided into 5 broad categories (Serban et al., 2013), namely: *expert systems*, *meta-learning systems*, *case-based reasoning systems*, *planning-based data analysis systems*, *workflow composition environments*.

Expert systems (ES) are the earliest and the simplest systems to provide help to the user during the data mining phase. Their main component is a knowledge base consisting of expert rules, which determine the mining algorithm to be used. Questions are posed to the user about a given problem and the metadata provided as response is used by the system in order to assess which rule is appropriate.

Meta-learning systems (MLS) are more advanced. The rules that were statically defined by the experts in the previous category are dynamically learned here. MLSs try to discover the relationship between measurable features of the dataset and the performance of different algorithms, which is a standard learning problem. The learned model can then be used to predict the most suitable algorithm for a given dataset.

The idea behind *case-based reasoning systems* (CBR) is to store the successfully applied workflows as cases, in a *case base*, with the only goal of reusing them in the future. When faced with a new problem (i.e., dataset) provided by the user, these systems return *k* previous cases from the case base according to the level of similarity with the new problem. The selected workflow can then be adapted to properly fit and solve the new problem. Their disadvantage, as in MLSs, is that they can provide structured help only if a new problem is similar to the problems seen so far.

Planning-based data analysis systems (PDAS) are able to autonomously design valid workflows without relying on the similarity between different problems. In order to do this, the workflow composition problem is seen as a planning problem, where a plan is built by combining operators that transform the initial problem into accurate models or predictions. In order to construct valid workflows, the input, output, preconditions, and effects of each operator need to be

known. Once the conditions are met, operators are composed to form valid but not necessarily optimal workflows, which at a later stage are ranked.
Workflow composition environments (WCE) do not provide automatic support for data analysis, but facilitate the use of different data mining algorithms providing nice graphical environments for quick workflow design and execution.

4 METADATA CHALLENGE IN KDD

In this section, we analyze what can be achieved by collecting metadata and what kinds of metadata can be collected in a KDD environment.

4.1 The Role of Metadata

The generation and management of metadata can determine the type of support offered. We differentiate among the following.

Single-step support. It is an indication of the complexity of the advice offered. The single step for which some kind of user support or even automation is provided is usually the data mining step of the KDD process.

Multi-step support. Similarly, it indicates the complexity of the advice offered. Metadata can be used to extend the support to several steps of KDD.

Variable selection support. It indicates whether a system provides user support in the very first phase of a KDD process. It is of crucial importance when an analysis of raw data needs to be done (e.g., in a Big Data environment). Raw data in this context refers to data that is not offered in a form of a dataset but, it is stored in its original format. Hence, prior to analysis, the data of interest needs to be selected and integrated into a unique dataset.

Explanations. It is easier for the user to design workflows when explanations are present. Explanations can be on operators for facilitating a design process as well as on results to help the user interpret them. This can be done by, for instance, giving useful instructions about statistical concepts.

Reuse of past experience. Metadata can increase reliability by enabling the reuse of workflows. The reuse of successful cases speeds up the process considerably. It allows to build on prior work and facilitates deeper analysis. It can enable truly collaborative knowledge discovery.

Automatic workflow generation. Metadata can drive the automatic composition and execution of the pre-processing and mining steps. This is the most advanced type of user support but at the same time the most challenging one.

Business understanding. Metadata can provide information about the meaning of the data, the terminology and business concepts and their relationships to the data. Metadata can provide information about the source of the data (provenance) and the path followed from a source to the current site (lineage).

4.2 Types of Metadata

The main objects participating in a KDD process include: (1) a *dataset* that needs to be analyzed, (2) *operators* used for pre-processing, and mining, as well as (3) *workflows*, which are combinations of operators with data in the form of directed acyclic graphs. In order to effectively support the user during the analysis, metadata should be stored for every aforementioned object. In addition, metadata that can boost the user support and which were not considered in this context are (4) *domain knowledge* used to store information for the concrete domain of data and (5) *lineage metadata*, relevant to justify the results of an analysis.

Metadata on the input dataset. The idea of characterizing a dataset has been researched from the early inception of meta learning. A dataset that needs to be analyzed - containing all the attributes that are relevant to the problem at hand - is assumed to be selected in advance and is generally described by the following groups of characteristics:

- *General measures:* include general information related to the dataset at hand. To a certain extent they are conceived to measure the complexity of the underlying problem. Some of them are: the number of instances, number of attributes, dataset dimensionality, ratio of missing values, etc.
- *Statistical and information-theoretic measures:* describe attribute statistics and class distributions of a dataset sample. They include different summary statistics per attribute like mean, standard deviation, etc.

However, if the problem to be solved is a prediction problem, then, a variable (or more) is defined to be a response variable. Once the response is defined, further metadata measuring the association between the remaining (input) variables and the response(s) (output) can be used to describe the dataset. Hence, we can additionally have the following groups of dataset characteristics:

- *Geometrical and topological measures:* this group tries to capture geometrical and topological complexity of class boundaries (Ho and Basu, 2002). It includes non-linearity, volume of overlap region, max. Fisher's discriminant ratio, fraction of instance on class boundary, ratio of avg. intra/inter class nearest neighbour distance, etc.

- *Landmarking and model-based measures:* this group is related to measures asserted with fast machine learning algorithms, so called *landmarkers*, and its derivative based on the learned models. It includes error rates and pairwise $1-p$ values obtained by landmarkers such as 1NN or DecisionStump as well as histogram weights learned by Relief or Support Vector Machines (SVM).

Metadata on Operators. They are typically expressed in the form of semantic information (e.g., ontology). By operators we mean all the different elements that can operate on a dataset. These include: (1) different transformation methods like normalization, discretization, etc., which are considered to be *pre-processing operators* and (2) different kinds of learning algorithms like decision trees, support vector machines, etc., which are considered to be *data mining operators*. Metadata on operators can be *internal* or *external* (Serban et al., 2013). External metadata treat an operator as a black-box, which means they only consider metadata with regard to the Input, Output, and some other properties like Preconditions and Effects (IOPE). Internal metadata tear up the box by considering metadata linked to an operator's internal *structure* (e.g., parameters or model type) or *performance* (e.g., speed, accuracy, model complexity).

Metadata on Workflows. The previously mentioned metadata are what systems need in order to provide assistance in terms of constructing valid workflows (e.g., all preconditions or input constraints of algorithms are met). However, the generated workflows may not necessarily be optimal. Moreover, the number of generated workflows can reach thousands, given the vast number of available data mining operators (e.g., Rapidminer, Weka). Thus, there needs to be a way of ranking the workflows. One way to do this is to keep track of metadata about workflows. In the eIDA system for instance, in order to characterize workflows, they follow a process mining-like approach. They extract generalized, relational, frequent patterns over the tree representations of the workflows (Kalousis et al., 2014).

Domain Knowledge. The effectiveness and need for domain knowledge in knowledge discovery has been confirmed in past research efforts. It is recognized by (Kopanas et al., 2002) that there is a role for domain knowledge in all stages of a KDD process. They demonstrate through examples how the domain expert is needed to (1) help define the problem by, e.g., giving business rules on what a failed transaction is or what is considered a problematic customer (2) assist in the creation of the target dataset by, e.g., defining the structure of the data and the semantic value of the data attribute values. However, in order to make use of it, domain knowledge should be represented by models that computers can understand. Ontologies are some of the successful knowledge engineering advances that can be used to build and use domain knowledge in a formal way. An ontology is an explicit specification of a conceptualization. Normally, it is developed to specify a particular domain (e.g., genetics). Such an ontology, often known as a domain ontology, formally specifies the concepts and relationships in that domain. Note that domain knowledge is not used by IDAs in the literature.

Lineage metadata. The KDD process can benefit from lineage metadata. Lineage metadata is composed of steps used to derive a particular dataset. It can be thought of as a recipe for creating data. The quality of the data for the user's analysis can be evaluated through the lineage of the dataset. Data quality of the source is important because errors introduced tend to inflate as the data propagates. This issue is even more critical when using raw data available in data lakes. The level of detail included in the lineage determines the extent to which the quality of the data can be assessed. If semantic knowledge of the pedigree is available, it is possible to automatically evaluate it based on quality metrics (Simmhan et al., 2005). All in all, this metadata can be used to understand and justify the results obtained during the analysis. This kind of metadata is also not considered in IDAs.

4.3 Comparison of Metadata on IDAs

In Table 1, we show types of metadata used by IDAs and types of provided support. For each cell in the table we put sign '+' if the system supports the particular concept described in the column and sign '-' if not. From the given table, we identify that many support limitations can be explained with the lack of proper metadata. Moreover, note that systems do not deal with the problem of variable selection (e.g., in a big data environment, provide support in terms of which variables are important to select for the analysis and combine them into a unique dataset) and none of the systems provides support in terms of business understanding. These limitations are due to the lack of appropriate metadata. We believe that domain knowledge and lineage metadata will improve the systems in this regard.

- ES and MLS do not use external metadata on operators (e.g., IOPE), therefore are not able to construct entire workflows.

Table 1: Type and role of metadata in IDAs.

Category	System	Metadata Type						Metadata Role						
		Input	Operator		Workflow	Domain Know.	Lineage	Single step supp.	Multistep supp.	Variable selection	Explanational	Reuse	Automation	Business underst.
			Int.	Ext.										
ES	SPRINGEX (Raes, 1992)	+	-	-	-	-	-	+	-	-	+	-	-	-
	MLT Consultant (Sleeman et al., 1995)	+	+	-	-	-	-	+	-	-	+	-	-	-
MLS	DMA (Giraud-Carrier, 2005)	+	+	-	-	-	-	+	-	-	-	-	-	-
	NOEMON (Kalousis and Hilario, 2001)	+	+	-	-	-	-	+	-	-	-	-	-	-
CBR	CITRUS (Engels, 1996)	+	+	-	+	-	-	-	+	-	+	+	+	-
	AST (Lindner and Studer, 1999)	+	+	-	-	-	-	+	-	-	-	+	+	-
	MiningMart (Morik and Scholz, 2002)	+	+	-	-	-	-	-	-	-	-	+	-	-
PDAS	RDM (Záková et al., 2011)	+	+	+	+	-	-	-	+	-	-	-	-	-
	KDDVM (Diamantini et al., 2009)	+	+	+	+	-	-	-	+	-	-	-	+	-
	eIDA (Kietz et al., 2014)	+	+	+	+	-	-	+	+	-	-	+	+	-
WCE	IBM SPSS Modeler	+	-	+	-	-	-	-	-	-	+	-	-	-
	SAS Enterprise Miner	+	-	+	-	-	-	-	-	-	+	-	-	-
	RapidMiner	+	-	+	-	-	-	-	-	-	+	-	-	-
	Weka	+	-	+	-	-	-	-	-	-	+	-	-	-

- MLS and CBR use huge number of input metadata but they do not provide support for automatically combining multiple steps.
- PDAS generate automatic workflows but they start from scratch every time. They do not make use of the experience from previous data analysis.
- WCEs allow to construct workflows but they do not provide much guidance.

5 METADATA CLASSIFICATION

The analysis in Section 4 showed that IDAs rely heavily on metadata in order to provide user support. In order to classify the identified metadata, we decided to extend the classification provided in (Foshay et al., 2007) and later extended in (Varga et al., 2014). Our classification can now capture the whole range of metadata required for the KDD process.

The classification tree is given in Figure 1. Note that the shaded shapes belong to the original classification that consists of the following metadata categories: *Definitional*, *Data quality*, *Navigational*, *Lineage*, and *Ratings*. Each category contains its respective metadata artifacts again denoted as shaded shapes in the figure. Nevertheless, in order to attach the required metadata artifacts, change and extension of the taxonomy was required, note the non shaded shapes. The imposed changes are the following: *Definitional* category is extended with a *Domain Knowledge* subcategory which is going to cover metadata related to the domain, *Data quality* is renamed to *Data characteristics* in order to better reflect the meaning of the participating artifacts. An additional category named *Activity characteristics* is added to capture active objects (e.g., operators) in a knowledge discovery process. An additional category *Assessment* is added with the aim of capturing the metadata artifacts with respect to the output of the knowledge discovery process. Next, the *Lineage* category is extended with three metadata artifacts discussed below. Moreover, additional artifacts belonging to different categories are further added.

For the purpose of our classification we clearly define all the categories and respective metadata artifacts below. Note however, that metadata artifacts that belong to (Foshay et al., 2007; Varga et al., 2014) are not discussed extensively. The interested reader is referred to those papers for further information.

The *definitional* category contains metadata that conveys the meaning of the data to the user or the system. From the original taxonomy in this category there are the integration *schema*, user *characteristics* and a *vocabulary* of business terminology. We extend the *definitional* category with the *domain knowledge* subcategory which is going to contain different metadata with regard to the domain. The idea is to enable a knowledge-rich data analysis. However, the goal of a knowledge-rich data analysis is not to provide a priori all the knowledge that might be required but to support a feedback loop by which a small amount of ini-

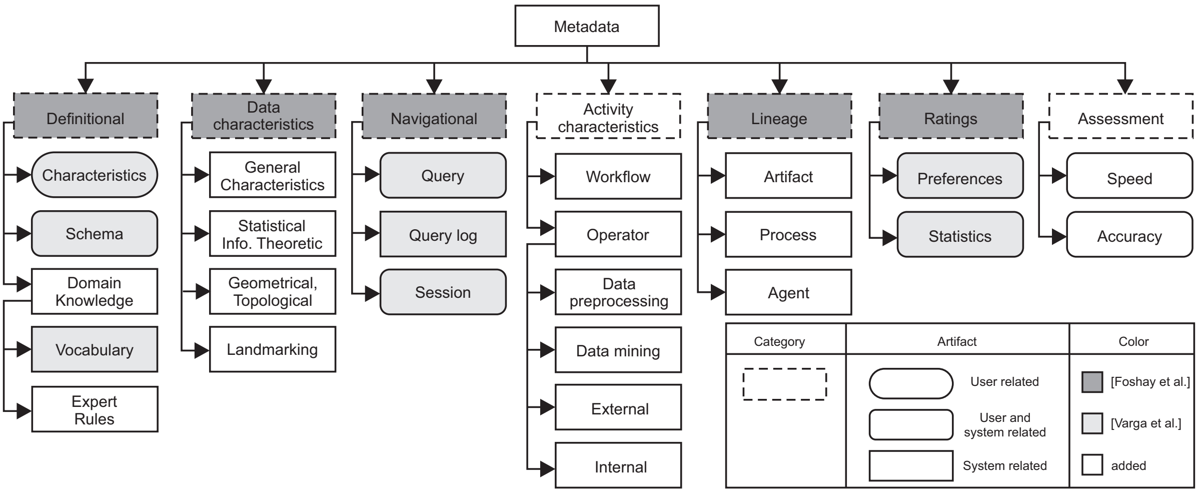

Figure 1: Metadata classification.

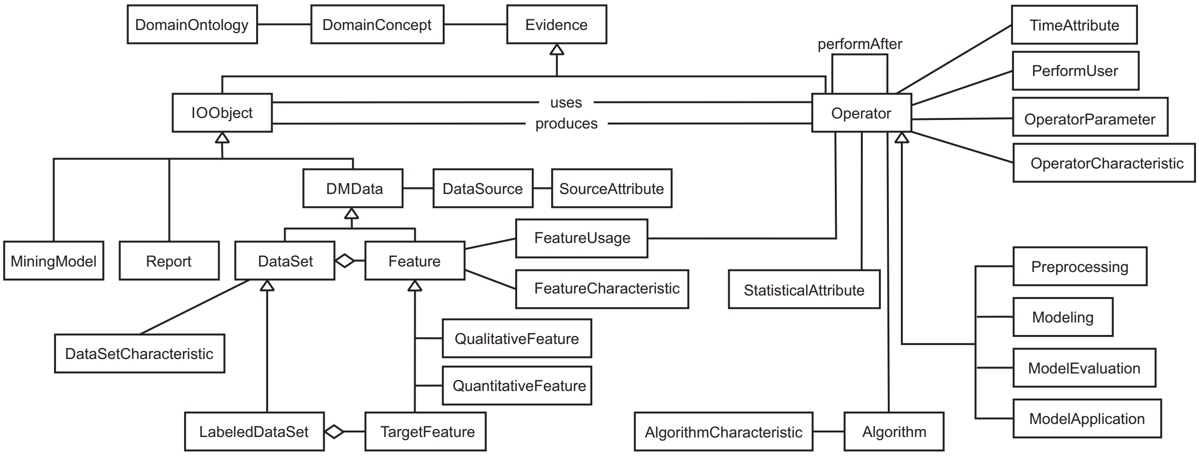

Figure 2: Conceptual schema of the metadata repository.

tial knowledge can be bootstrapped into more knowledge by mining, which can in turn be complemented by more human-supplied knowledge to allow further mining, etc. Hence, under the domain knowledge we place the *vocabulary* artifact from the original classification, this can be replaced or can easily represent the *domain ontology* discussed in Section 4.2. Furthermore, we add *expert rules* as metadata which can represent the expert knowledge for the domain.

Data characteristics consists of artifacts that convey information about the characteristics of data that are of crucial importance to a knowledge discovery process. They advice the system about the completeness or even validity of data. Metadata artifacts in this category are those detected in the analysis in Section 4.2.

The *navigational* category comes from the original classification and keeps track of how the user explores and navigates through data. The metadata artifacts considered under this category can be useful for enabling user support in a data selection phase prior to data mining (e.g., suggesting the user relevant attributes using past experience). Metadata artifacts are: *query*, *query log*, and *sessions*.

The *activity characteristics* category consists of metadata artifacts whose expressiveness determines the degree of automation that can be achieved in the process of knowledge discovery. These are the most important metadata required in a KDD process. Note that these kind of metadata were not considered in the previous classifications. There are two main metadata artifacts considered here, namely metadata on *operators* and metadata on *workflows* (see Section 4.2).

Lineage consists of artifacts that model resources (e.g., data-sets) as *artifacts*, *processes* (e.g., actions or series of actions performed in artifacts or caused by artifacts, and resulting in new artifacts) and *agents* (e.g., contextual entities acting as catalysts of a process, enabling, facilitating, controlling, or affecting

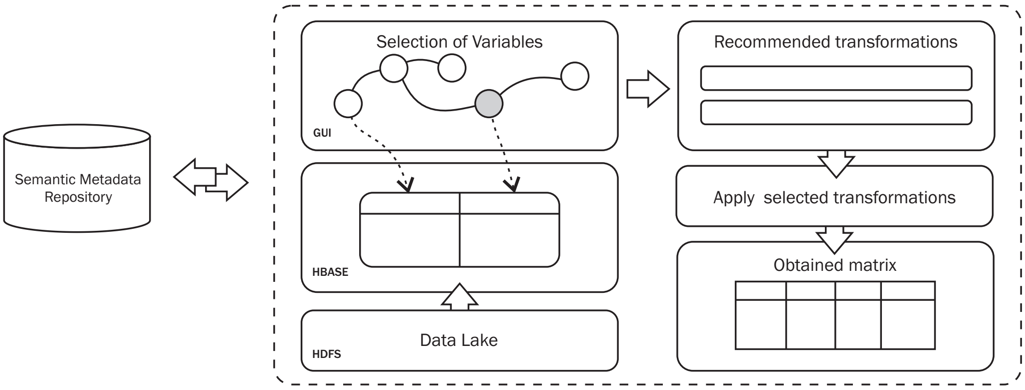
Semantic Metadata Repository
Selection of Variables
GUI
HBASE
HDFS
Data Lake
Recommended transformations
Apply selected transformations
Obtained matrix

Figure 3: High level view of the proposed system.

its execution) (Moreau et al., 2011). The aim of lineage metadata is to capture the causal dependencies between the artifacts, processes, and agents.

The *Ratings* category comes from the original taxonomy and it contains metadata such as user *preferences* and usage *statistics*. However, note that the *preferences* artifact is important with regard to knowledge discovery as well. It can store different user goals, which can be used by the system to design workflows optimizing some performance measure associated with the user goal. Finally, *statistics* relates to the data usage indicators. It can keep evidence of which data are explored more.

The *Assessment* category consists of metadata artifacts with regard to the output of a knowledge discovery process. They can be used to assess how good are the generated DM workflows. This is defined by the correctness in execution and its performance with respect to evaluation criteria, such as *accuracy* and *speed*. These metadata can be used to list the best performing workflow or rank all of the constructed workflows.

6 METADATA REPOSITORY

After having identified the metadata required, we turn on discussing how these metadata can be stored and managed.

The best way to store metadata is to store them in a metadata repository. However, usually metadata remain hidden in scripts and programs, without being further reused. This is also what we realized is happening in practice in a project we are developing with a multinational company located in Barcelona[1].

The project aims at improving the current state of the data analytics process in the company. The idea is to allow data analysts to easily select relevant variables for their analysis and assist them during the data pre-processing and mining. The company stores the variables or the data in a raw format in a Data Lake in a Hadoop ecosystem. In order to allow an easy selection of variables and provide user support during the pre-processing phase (e.g., recommend pre-processing operations particularly suited for the domain) we created a semantic repository with the aim of storing all the necessary metadata. The variables in the Data Lake and their respective characteristics will be mapped to corresponding concepts in the repository. In addition, different possible transformations (pre-processing operations; domain knowledge) will be described in the repository and they will be linked to corresponding concepts. The user will be able to easily access the variables through the graphical interface which is going to be fed by the repository. After selecting the variables (e.g., their corresponding concepts) of interest proper transformations will be recommended. The information of which pre-processing will be applied to a given variable will be deduced from the metadata repository. Hence, not everybody in the need of analyzing the data will have to be an expert of the domain, as happened to be the case previously in the company. Domain specific knowledge will be added once to the repository, and will be used automatically (repeatedly) by everyone wishing to analyze the data. A high level architecture of the system proposed for the project is shown in Figure 3.

The software components accessing the repository will be ”bound” to the given metadata structure which is conceptually described by a schema shown in Figure 2. The comprehensive schema proposed in this paper proves to be useful in the project.

The schema can be logically divided into three main parts. The first keeps track of the domain knowledge, the second manages information with regard to passive elements, and they fall under the *IOObject*

[1] https://inlab.fib.upc.edu/en/big-data-analytics-lab

class, and the third manages information with regard to active elements and they fall under the *Operator* class.

Implementation. We used Resource Description Framework (RDF) as a data model for storing the metadata. In RDF, statements about resources can be made in the form subject-predicate-object expressions and they are called triples. Hence, our repository is defined as a triple store, where we used OpenLink Virtuoso as a storage engine. The repository is provided as a Web Service and an application for metadata management is built on top of it. JavaServer Pages (JSP), Asynchronous JavaScript (AJAX) and XML are used to implement the application and the graphical user interface.

7 CONCLUSION AND FUTURE WORK

The process of knowledge discovery is challenging. Data relevant to the analysis needs to be selected, preprocessed, mined and finally evaluated. Beginners are alarmed by the myriad of operators and more experienced users limit their activity to several known approaches. A thorough user assistance is necessary. Therefore, systems with the aim of assisting the user during this process are built. We studied these systems with the goal of identifying the metadata used to enable the assistance. Hence, we identified the metadata used to provide user support during the KDD process. We found out that important metadata such as domain knowledge and lineage which can make the life of a data analyst easy, have not been considered. We provided a classification of the metadata found. We proposed a comprehensive metadata framework that captures the complete range of metadata needed to assist the user during the whole process of KDD. We showed the importance of such metadata in a real project by implementing a metadata repository to store and manage the whole range of metadata.

In our future work, we are planning to extend the domain knowledge incorporated into the repository and we are planning to develop tools for exploiting the metadata. We are going to test different ways of reasoning on top of the metadata. Moreover, we will be exploring the idea of incorporating meta learning into the whole picture.

ACKNOWLEDGMENTS

This research has been funded by the European Commission through the Erasmus Mundus Joint Doctorate "Information Technologies for Business Intelligence - Doctoral College" (IT4BI-DC).

REFERENCES

Bernstein, A., Provost, F. J., and Hill, S. (2005). Toward intelligent assistance for a data mining process: An ontology-based approach for cost-sensitive classification. *IEEE TKDE*, 17(4).

CWM (2003). Object Management Group: Common warehouse metamodel specification. Available at http://www.omg.org/spec/CWM/1.1/PDF/.

Diamantini, C., Potena, D., and Storti, E. (2009). Ontology-driven KDD process composition. In *IDA*.

Engels, R. (1996). Planning tasks for KDD; performing task-oriented user-guidance. In *KDD*.

Fayyad, U. M. et al. (1996). From data mining to knowledge discovery in databases. *AI Magazine*, 17(3).

Foshay, N. et al. (2007). Does data warehouse end-user metadata add value? *Commun. ACM*, 50(11).

Giraud-Carrier, C. (2005). The data mining advisor: meta-learning at the service of practitioners. In *ICMLA*.

Ho, T. K. and Basu, M. (2002). Complexity measures of supervised classification problems. *IEEE TPAMI*, 24(3).

Kalousis, A. et al. (2014). Using meta-mining to support DM workflow planning and optimization. *JAIR*, 51(1).

Kalousis, A. and Hilario, M. (2001). Model selection via meta-learning: A comparative study. *IJAIT*, 10(4).

Kietz, J., Serban, F., Fischer, S., and Bernstein, A. (2014). Semantics Inside! But Let's Not Tell the Data Miners: Intelligent Support for Data Mining. In *ESWC*.

Kopanas, I., Avouris, N. M., and Daskalaki, S. (2002). The role of domain knowledge in a large scale data mining project. In *SETN*.

Lindner, G. and Studer, R. (1999). AST: support for algorithm selection with a CBR approach. In *PKDD*.

Moreau, L. et al. (2011). The open provenance model core specification (v1.1). *FGCS*, 27(6).

Morik, K. and Scholz, M. (2002). The miningmart approach. In *Informatik bewegt: Informatik*.

Raes, J. (1992). Inside two commercially available statistical expert systems. *Statistics and Computing*, 2(2).

Serban, F., Vanschoren, J., Kietz, J., and Bernstein, A. (2013). A survey of intelligent assistants for data analysis. *ACM Comput. Surv.*, 45(3).

Simmhan, Y. L., Plale, B., and Gannon, D. (2005). A survey of data provenance in e-science. *SIGMOD Rec.*, 34(3).

Sleeman, D. H., Rissakis, M., Craw, S., Graner, N., and Sharma, S. (1995). Consultant-2: pre- and post-processing of ML applications. *IJHCS*, 43(1).

Varga, J. et al. (2014). Towards next generation BI systems: The analytical metadata challenge. In *DaWaK*.

Záková, M., Kremen, P., Zelezný, F., and Lavrac, N. (2011). Automating KD workflow composition through ontology-based planning. *IEEE T-ASE*, 8(2).