SISTEMAS DE INFORMAÇÃO
Revista de Sistemas de Informacao da FSMA n. 8 (2011) pp. 36-45 http://www.fsma.edu.br/si/sistemas.html

# Meta-DM: An ontology for the data mining domain

Edmar Augusto Yokome e Flávia Linhalis Arantes

**Abstract – Ontologies have been used in various research areas from computer science, including data mining. This article describes the development of a domain ontology for data mining. Meta-DM ontology provides a common terminology that can be shared and processed by data mining tools. The ontology also indicates the KDD phases where human knowledge is necessary - this is its greatest differential. With ontologies as the Meta-DM, it is possible to add semantics in the data mining process in order to improve the interaction and cooperation between experts and mining systems.**

**Keywords – domain ontology, data mining, knowledge discovery in databases (KDD)**

## I. Introduction

In Artificial Intelligence, an ontology may be defined as an "explicit and fol specification of a shared concept on a field of interest " [1]. Ontologies are often used as structures that represent the knowledge on a specific areas (or domain) through relevant concepts and the relationship among them. Using ontologies, knowledge representation on a specific domain can be more easily understood and shared among human and software agents.

Nowadays, ontologies are used in several domains, such as semantic web, databases, expert systems, and others. In the field of data mining, using ontologies in projects that involve the discovery of knowledge in databases is still little explored in literature [2].

The process of Knowledge Discovery in Databases consists in discovering interesting information in out databases. This information can be used to improve or solve a specific problem, such as improving a supermarket marketing campaign, help decide the best moment to buy or sell stock in the stock market, etc. Its usage is justified by the huge growth in databases in the last few years and by the fact that all data sources have become more heterogeneous (relational databases, time series, multimedia, images and others) rendering manual analysis impossible.

KDD is performed in different phases: cleaning, integration, selection, transformation, mining, evaluation and presentation. Data mining tools, such as Weka [3], are purely data centric, that is, they work with data separately in each phase of KDD, allowing for a human expert to guide the process of knowledge discovery in databases.

The KDD process can benefit from ontology oriented approaches, in which semantic can be inserted in the mining process in order to help and guide the data miner during the process of knowledge discovery. Data mining tools, when combined with problem domain knowledge represented in ontologies, can help the data miner in tasks such as understanding and preparing the data, relevant data selection, restriction specification to guide the choice of mining algorithms and others [4].

In this paper we present the development of a domain ontology to the field of data mining. Meta-DM ontology offers a common terminology for data mining and can be used with several tools applied to this domain, helping the data miner during the process of knowledge discovery in databases.

In order to make it feasible to effectively help the data miner during KDD, data mining tools must include human and domain knowledge, so that the discovery of knowldge happens interactively [5], [6]. Therefore, Meta-DM ontology intends to identify the concepts and relationships where human knowledge used to understand the problem is necessary during KDD.

When using a domain ontology such as Meta-DM to guide the KDD, the data miner will have a set of semantic items that will help him through the many phases of this process, in order to find more interesting results.

This article is organized as following. In the section II we present and justifiy the theme choice. In section III we present them problem that motivated this work, as well as the adopted solution. In section IV we present a brief theoretical foundation with the concepts on data mining and otologies that are relevant to this work. In section V we present some papers related to the theme and we compare their proposals with Meta-DM ontology. In section VI we present the methodology used to develop the ontology. In section VII we present the ontology development process, detailing the task performed at each phase of its life cycle. In section VIII we present some pointer for future works in our research. At last, in section IX we present the conclusions of this paper and the ontology Meta-DM as the main result of our result up to this moment.

## II. Justification

According to Cao and Zhang [6], data mining is practiced nowadays as an automatized process that provides algorithms and tools, with little human involvement and without the ability to adapt the process to the restrictions of the environment (context domain). The consequence of this fact is that the results of data mining sometimes are not interesting to the business goals toward which the data mining project was developed. Cao and Zhang proposed a methodology called $D^3M$, that considers human knowledge and context information on the problem at hand during data mining. The $D^3M$ methodology has the following characteristics:

- Context based restriction: to deeply know the environment around the problem domain, its data and goals.

Possible ways to have context based restrictions are through the usage of domain metadata or mining problem related ontologies.

- Domain knowledge integration: relates to the way knowledge on the probem domain can be represented and integrated to the process of database knowledge discovery. Using ontologies related to the problem domain is one the approaches that are adequate to model and integrate domain knowledge to the data mining process.
- Cooperation between man and machine: allowing for cooperation between experts and mining system throughout the whole process.
- Depth mining: consists in evaluating and refining the triggerable rules. Before starting the data mining, the miner may define which rules are according o the business interests. These rules, called "triggerable", may be triggered during the mining process.
- Improve knowledge "actionability": generic patterns may need improvements to generate actionable patterns, that is, patterns that are according to the interests of the mining problem at hand.
- Interactive result refinement process:0valuation and refinement of the results are based on interactive feedback until the final phase.
- Support to interactive and parallel mining: consists of getting user requests, managing information and using algorithms to process them in several machines.

This work is justified by the current trend in data mining that is the insertion of human and domain knowledge during the process of knowledge discovery in data bases. We hope to contribute to the state of the art through the development of a data mining domain ontology so that we can insert context and human knowledge in order to allow the mining tools to help interactively the data miner during the knowledge discovery process.

## III. Problem and Proposed Solution

Data mining, as practiced today, is highly data oriented, that is, there is little interaction with the data miner during the knowledge discovery process [6]. Because of that, many times the results achieved are not interesting for the problem at hand (business goals). This is consequence of the lack of semantics and of greater interaction between the miner and the data mining tools. As previously stated, newer methodologies, such as $D^3M$, have arisen to solve this problem and insert human and domain knowledge into the mining process.

The problem to be solved through our research is how to insert human and context knowledge related to the mining problem at hand into data mining tools so that they can work interactively with the miner during the KDD process. In order to contribute to the solution of that problem we developed an ontology for the data mining domain. Therefore, in this paper, we propose Meta-DM, an ontology for data mining tools, in order to identify points where human knowledge is necessary during the KDD process, and, therefore, where domain semantics should be inserted.

There is some previous work in the literature related to the development of ontologies to the data mining domain, as we present in V. Nevertheless, none of them concerns specifically the development of ontologies for data mining tools, identifying points where human knowledge is necessary, so that mining tools can work collaboratively and interactively with the miner during the knowledge discovery process.

## IV. Theoretical Foundations

The theoretical foundation of this paper involves two areas of Computer Science, ontologies and the discover of knowledge in databases. The subsections that follow will describe both of them briefly.

### A. Ontologies

An ontology is a form to explicitly and formally define the concepts and restrictions related to a specific domain [7]. They are used to represent knowledge on a specific area or domain through the definition of concepts and relationships.

We use methodologies to discipline and support the creation of ontologies, such as METHONTOLOGY [8], Noy e McGuiness [9], Grüninger e Fox [10], Uschold e King [11], and others. To develop the ontology described in this article, we used the methodology METHONTOLOGY [8], as presented in the section VI. This methodology is based on the construction of ontologies from scratch but allows for the reuse of other ontologies. The choice of this methodology was due to the fact that its phases are distinct and well documented, making it easy to learn for a beginner in the field of ontology development.

To formalize ontologies many different languages can be used, such as OWL (Web Ontology Language) [12], [13], Ontolingua [14], LOOM [15], F-Logic [16], etc. In this work, the semantic web language OWL was used. This language is used together with RDF and RDF-S, which are also recommendations from the W3C to the semantic web.

OWL is used to represent explicitly the meaning of terms in vocabularies and in relationships among them [13], RDF is a language to represent information on resources [17] and RDF Schema is a semantic extension to RDF, that provides mechanisms to describe resource groups and the relation between the resources that make them [18].

Among the reasons that lead us to choose OWL, we can say that it is a W3C recommendation since 2004 and is considered a *de facto* standard for the development of ontologies, and its usage can make it easier to integrate Meta-DM with applications developed in the context of the Semantic Web.

Tools to develop ontologies are very useful, for they increase productivity and provide several resources that make it easier to develop ontologies. Among the tools available freely we have Protégé [19], Ontolingua Server [14], KAON [20], SWOOP [21], and others.

The tool chosen for the development of Meta-DM was Protégé [19]. Among its advantages we have good support to OWL, good documentation (including tutorials), it is open source, has an active user and developer community and is, therefore, widely used by several research groups. Besides, Protégé offers a large set of plugins that can be added according to the needs of the application and/or the ontology.

Among the plugins we used we had OWLViz [22] to visualize graphically the development of the ontology Pellet [23] to verify the consistency among the classes declared in the ontology.

### B. Knowledge Discovery in Databases

The process of Knowledge Discovery in Databases (KDD) is a branch of Computer Science whose goal is to find interesting patterns in data bases. The KDD process can be seen in Figure 1 and is composed of the following phases: cleaning and integration, selection and transformation, mining, evaluation, presentation and knowledge. Data mining is one of the phases of KDD where knowledge is mined or extracted from a huge pile of data [24].

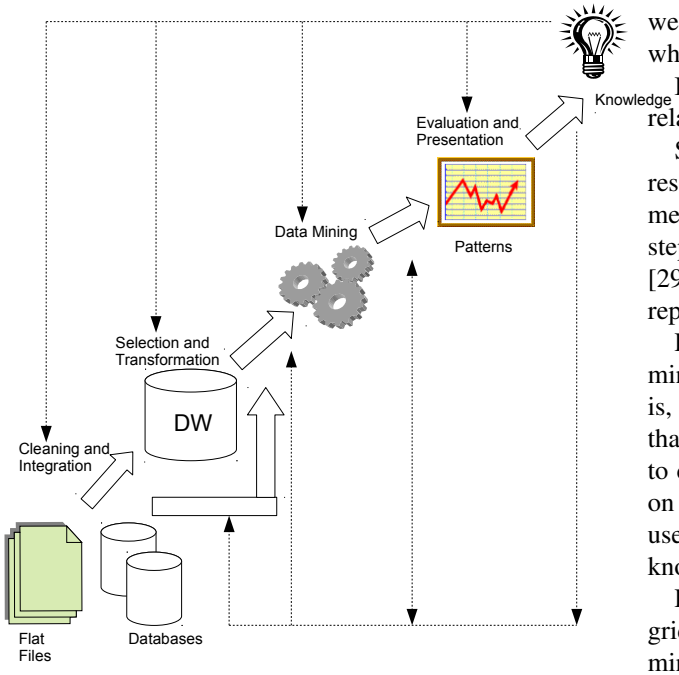


Fig. 1. Phases of the process of Knowledge Discovery in Databases (KDD) [24].

In order to help developing data mining projects, there are several methodologies that discipline and help designer. Among them, CRISP-DM and $D^3$M were the ones chosen for this work.

CRISP-DM is a data oriented methodology that includes all KDD phases [25]. It is composed of six steps: business understanding, data understanding, data preparation, modeling, evaluation and application. This methodology was very important in the definition of all concepts used in the Meta-DM ontology.

$D^3$M was developed with the goal of allowing data mining to be more interactive, taking into account the knowledge of the problem domain during the execution of a data mining project [6]. This methodology was important to identify the points where human knowledge is necessary for the understanding of the problem during the KDD process.

Using data mining tools during a KDD project allows for more efficient results because the size and diversity of database sources makes it unfeasible for a human being to perform data mining without an adequate tool. As examples of tools we have Kira [26], Weka [3], Tanagra [27], Oracle Data Mining [28], and others. Weka and Kira were studied in order to know the terminology used in this kind of tools, the former because it is widely used and the latter because it is the result of a research project whose goal is to guide the miner during data mining projects (please see further explanation in section VIII).

## V. Related Works

In order to develop the Meta-DM ontology, we researched several other existing works in literature on the data mining domain. These works served as a first guide to develop our ideas and to allow us to understand the extra contribution that we could give when developing the Meta-DM ontology and what amount of previous work could be reused.

In this section, we are going to present some of the main related works compared to Meta-DM.

Sharma and Osei-Bryson's ontology was developed to represent the business understanding phase of the CRISP-DM methodology [25]. The authors raised issue related to this step in order to help the miner in understanding the business [29]. Unlike Sharma and Osei-Bryson's, Meta-DM intends to represent all the steps in a data mining project.

DM Ontology [30] is an ontology that contemplates data mining applied to marketing, specially the financing part. That is, this ontology was developed considering a specific problem that data mining will help solving. The goal of Meta-DM is to contemplate all steps of data mining, without concentration on a specific problem. Nevertheless, other ontologies may be used together with Meta-DM in order to provide the domain knowledge necessary to solve a specific problem.

DMO ontology [31] was developed in order to guide a grid data mining project. Its major goal is to perform data mining using semantic services spread through the web. To achieve that, it used OWL-S ontology to describe semantic web services. The ontology uses concepts of KDD phases, but has different goals than Meta-DM.

Pinto and Santos' ontology [4] used some concepts of DMO [31] and was developed in order to contemplate exclusively the KDD phases following the METHONTOLOGY methodology [8]. Some concepts of the Pinto and Santos' ontology were used in Meta-DM. This ontology was the one closest to our proposal, a fact that justifies a closer attention to this work. Nevertheless, Meta-DM ontology also takes into consideration the CRISP-DM methodology [25] and intends to be an ontology for data mining tools, also identifying the moments where human knowledge on the problem is necessary, in order to increase the cooperation and interactivity between miner and mining tools. This fact justifies the creation of Meta-DM instead of just using the Pinto and Santos' ontology.

The Exposé ontology [32] was created to record machine learning experiments and the work flow for the knowledge discovery process, where this information could be collected, shared and reused, using a common vocabulary on data mining

and the choice of algorithms and data structures. The Exposé ontology works in a network and uses other data mining domain ontologies. its goal is to record data mining experiments and to share these experiments with other applications. The Meta-DM was built in order to generically guide data mining tools during the process of KDD. Other ontologies related to the mining problem domain may be used together with Meta-DM in order to supply context information. Nevertheless, Meta-DM does not have the primary goal using other data mining domain ontologies in its applications.

The OntoDM ontology [33] intends to create a set of term definitions for the data mining domain, such as data sets, data mining tasks, data mining algorithms, and others. This way, ontology development projects for this domain could use some of its definitions, avoiding interpretation ambiguities in any point of the domain.

In 2009, OntoDM ontology [34] was updated and new data mining base entities definitions were added, such as restricting a domain context or descbring different aspects of data mining. According to Panov et al. [35], the ontology is based in a framework that represents the entities in a data mining project. As stated above, OntoDM intends to unify the data mining domain a nd to formalize definitions and results achieved through data mining. This ontology is classified as a heavyweight, whose goal is to represent all componets belonging to a mining project and offer a common terminology to all data mining projects.

Unlike OntoDM, Meta-DM intends to be a lightweight ontology approaching the entire data mining process but without going deep into any of the phases. Since this deepening depends partly on the mining problem at hand, Meta-DM can be specialized or used together with a problem domain ontology, as described int he section VIII. The goal of serving as a common terminology for the entire data mining domain is contemplated by both ontologies but OntoDM does not represent formally the need for human knowledge in the KDD process, which becomes the major differential of Meta-DM when compared to other previously defined ontologies. This allows for data mining tools to add interfaces that can make it easer to cooperate and interact with human experts.

## VI. Methodology for the development of Meta-DM

The METHONTOLOGY methodology [8] was used to develop Meta-DM. This methodology is based on construction of ontologies from scratch, using other ontologies or not. The authors compare the life cycle of an ontology with the life cycle of tradition software and stress that is quite complicated to discover all necessary requisites before starting development.

The phases for the life cycle of an ontology are:, knowledge acquisition, conceptualization, integration, implementation, evaluation and documentation (see Figure 2). Now we will describe briefly each of those phases.

1) Specification: The goal of this phase is to create a document containing the ontology's specification, written in natural language, using an intermediate representation set or competence issues [10]. In this phase it is proposed that at least the following information is included:
   - Purpose: including users, use case, final users, etc.
   - Fomality level of the implemented ontology, which can be highly formal, semi-fmral ou rigorously formal.
   - Scope of the ontology: includes a set of terms to be represented, its characteristics and granularity.
2) Knowledge acquisition: This phase is performed simultaneously with the specification and is concerned with the acquisition of the body of knowledge that is necessary to start the ontology creation process. In order to learn everything required you may need to consult experts, books, manuals, figures, tables and even other ontologies and everything may be put together with techniques such as brainstorming, interviews, formal and informal text analysis and knowledge acquisition tools.
3) Conceptualization: in this phase the domain knowledge will be structured in a conceptual model that will describe the problem and its solutions according to a domain vocabulary identified in the ontology specification activity. The first thing to do is gather a complete Glossary of Terms (concepts, instances, verbs and properties), that will summarize everything that is useful and may be potentially used in the domain knowledge and its meaning. Once the glossary is completed, we must group the terms in concepts (data dictionary that describes the concepts, their meanings, attributes and instances) and verbs (domain actions). In the end of this phase we will have created a conceptual model expressed as set of well defined terms that will allow the final user verify whether the ontology will be useful for the application without inspecting its source code and compare the scope and plenitude of other ontologies, its reusability and the compatibility given the knowledge analysis.
4) Integration: To speed up the development process, one may consider reusing a set of definitions already developed inside other ontologies instead of starting from scratch.
5) Implementation: Consists of implementing the ontology in a formal language such as OWL [13], Ontolingua

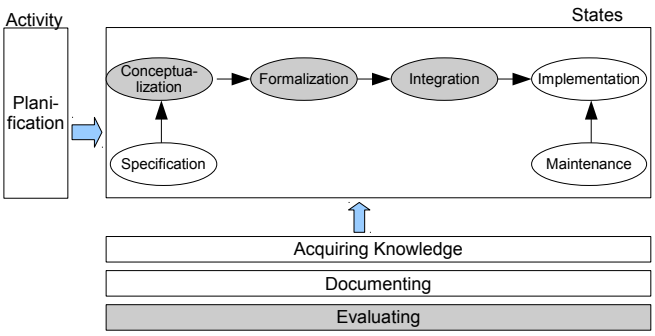


Fig. 2. Phases of METHONTOLOGY methodology, which was used in the development of the Meta-DM ontology. The phases of knowledge acquisition, documentation and evaluation are present in the whole development process [8].

[14], LOOM [15], F-Logic [16], or other. In this phase it is required an ontology development framework that must include at least lexical and syntactic analysis, an editor, a navigator, a search tools that can look for terms and present the results found.

6) Evaluation: Perform a technical judgment of the ontology, its software environment and the documentation of all phases of its life cycle. The evaluation includes verification (the technical process to assure correctness) and validation (the process to assure that the ontology effectively represents the knowledge domain defined in the specification phase).
7) Documentation: For each of the previous phases a document describing what was performed is written. Documentation is an integral part of the ontology development. Hence, this step is a part of all the previous ones.

This set of steps is the life cycle for the development of an ontology. This methodology was adopted in our research to register each of the steps in the development of Meta-DM.

## VII. Development of the Meta-DM ontology

Meta-DM intends to guide de KDD process and identify the points where human knowledge on the mining problem is necessary. Using it, we intend to insert process semantics on the mining process and allow for the participation of the miner in the database knowledge discovery process.

Meta-DM was build based on the methodology CRISP-DM, which was used to identify all the steps of the data mining process and represent them in the ontology. We used them Weka and Kira tools in order to identify the tasks that should belong in Meta-DM.

Meta-DM is presented in Figure 3 where it is shown that it is composed of ellipsis (representing a concept), arrows (representing a relationship), lines (representing a dependence), a connection with a filled circle (representing an attribute) and a dotted arrow (representing a point when human knowledge is necessary).

Figure 2 shoed the phases of the life cycle of Meta-DM development. The following subsections will show in detail how we performed each of using the methodology METHONTOLOGY.

### A. Specification

In this phase, domain and scope of the ontology were determined, that is, we defined that the ontology would represent the data mining domain, with the goal of guiding the data mining process inside mining tools. We also defined that the CRISP-DM methodology should be taken into account in order to perform all the steps that make a data mining project.

In this phase, we also defined that the ontology should include the moments when there is need of human knowledge related to the problem during the mining process.

### B. Knowledge Acquisition

In order to develop a domain ontology, one needs to know very well the domain at hand, that is, to become and expert in the field. Before starting the development of Meta-DM, it was necessary to study data mining. For that the following resources where used: books, classes, practical assignments, articles and help from experts in the field.

Studying data mining was important to know what concepts should be included in the ontology proposed and to understand how they related.

### C. Integration

We consider reusing previously existing ontologies when developing Meta-DM. Several other ontologies were analyzed, as described in section V.

The classes `Algorithm` and `Data`, as well as its subclasses, were reused from Pinto and Santos' ontology, as the next section will show.

### D. Conceptualization

In the conceptualization phase, we enumerated the important terms for the ontology and defined the classes and hierarchy among them. The classes and their relationships were represented in the Protégé tool.

The diagram of Meta-DM is presented in Figure 3. Among the elements that compose it, we have classes, relationships and attributes. The development was performed in cycles, where new concepts and relationships were identified as the ontology evolved, until we came to the result shown in Figure 3.

The ontology also represents through a dotted line the relationships where human knowledge (or expert intervention) is needed in order to discover knowledge in data bases. There is, therefore, two types of relationships between elements in the ontology – the ones that involve human knowledge (dotted lines) and the ones that are automatic in the KDD process (continuous lines).

The ontology can be divided in five parts: data, problem understanding, data treatment, mining task and results. The concepts of the ontology which relate to each of those parts in described in the following subsections.

*1) Data:* The class `Data` is responsible for holding all the data and their structure. Its subclasses, – `Value` e `Structure` – specify values and the database structure.

*2) Problem Understanding:* The mining problem must be analyzed and understood as related to data (`Data Understanding`) and as related to the mining business (`Business Understanding`). The latter is done according to the goal definition (`Objective`) and to problem description (`Problem`) that the mining project needs to solve. It should be noted that in this phase of data mining, human knowledge on the mining problem at hand is necessary, what is represented in the ontology through dotted lines.

*3) Data treatment:* This part of the ontology represents the database in a way that is adequate to the data mining project. The main class is `Table for Analyis`, which represents data ready to go through the mining process. The relationships `hasIntegration`, `hasTransformation`, `hasSelection` and `hasCleaning` representa ctions that

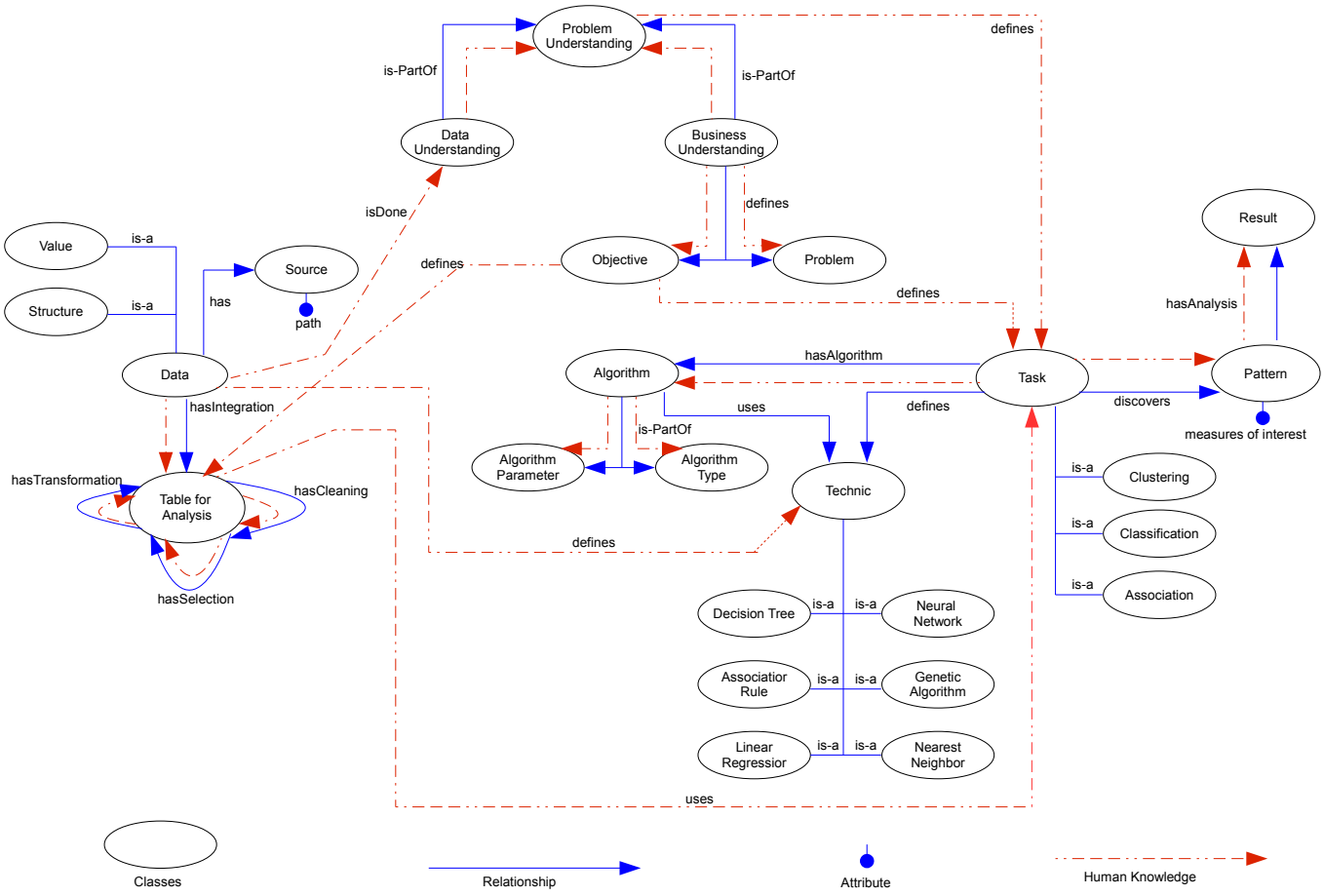


Fig. 3. Diagram of Meta-DM showing classes, relationships, attributes and points where human knowledge is necessary during the KDD process.

need to be performed to treat the database and respectively represent database integration, database transformation, relevant data selection and cleaning of "noisy" or inconsistent data. In all those phases, the knowledge of an expert on the mining problem at hand is necessary.

*4) Mining Task:* Represents the definition of the data mining task through the class `Task` and its subclasses. Together with the data mining task, it is necesary to define the technique used (classe `Technic` e suas subclasses) e seu respectivo algoritmo (class `Algorithm` and its subclasses). In these phases, human knowledge is also necessary.

The mining task is predefined with the business and data understanding. In this phase it is defined which task better suits the data mining project goals. Defining the task we also define one or more techniques according the data types used by a specific mining algorithm.

*5) Results:* The last part of the ontology represents the patterns found after the execution of the data mining algorithm (class `Pattern`) and the results found (class `Result`). Human knowledge is required to interpret those results and describe them in order to verify whether or not the mining project achieves the goals toward which it was created.

### *E. Formalization/Implementation*

Formalization was made inside the Protégé tool, using the OWL language. The fragment of code presented in this section, serialized in Turtle [36], shows the class `Task` code, its subclasses and the relationships that leave the class `Task` and arrive at the classes `Pattern`, `Technic` and `Algoritm`. Figure 4 shows the part of the ontology that is represented in the OWL code fragment.

The other parts of Meta-DM (classes and relationships) were formalized in a similar way and may be found in [37].

```
01  :Task rdf:type owl:Class .
02
03  :Technic rdf:type owl:Class .
04
05  :Pattern rdf:type owl:Class .
06
07  :Algorithm rdf:type owl:Class .
08
09  :Association rdf:type owl:Class ;
10      rdfs:subClassOf :Task .
11
12  :Classification rdf:type owl:Class ;
13      rdfs:subClassOf :Task .
14
15  :Clustering rdf:type owl:Class ;
16      rdfs:subClassOf :Task .
17
```

```
18  :discovers rdf:type owl:ObjectProperty ;
19     rdfs:domain :Task ;
20     rdfs:range :Pattern ;
21     rdfs:subPropertyOf owl:topObjectProperty .
22
23  :discovers_human rdf:type owl:ObjectProperty ;
24      rdfs:domain :Task ;
25      rdfs:range :Pattern ;
26      rdfs:subPropertyOf owl:topObjectProperty .
27
28  :hasAlgorithm rdf:type owl:ObjectProperty ;
29     rdfs:domain :Task ;
30     rdfs:range :Algorithm ;
31     rdfs:subPropertyOf owl:topObjectProperty .
32
33  :hasAlgorithm_human rdf:type owl:ObjectProperty ;
34     rdfs:domain :Task ;
35     rdfs:range :Algorithm ;
36     rdfs:subPropertyOf owl:topObjectProperty .
37
38  :defines rdf:type owl:ObjectProperty ;
39      rdfs:domain :Task ;
40      rdfs:range :Technic ;
41      rdfs:subPropertyOf owl:topObjectProperty .
```

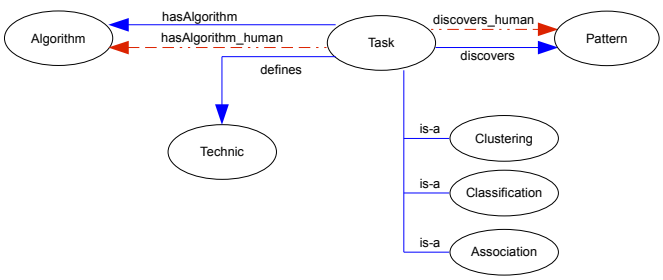


Fig. 4. Fragment of the Meta-DM ontology showing the class Task, its subclasses and its relationships with the classes Pattern, Technique and Algorithm.

Lines 01 through 07 show the creation of classes `Task`, `Technic`, `Pattern` and `Algorithm`. These classes are concepts of the ontology that represent, respectively, the mining task, the mining technique that we will work with, the patterns generated and the data mining algorithm to be used.

Lines 09 through 16 create the classes `Association`, `Classification` and `Clustering`as subclasses of `Task`. These subclasses represent the base tasks of data mining.

As mentioned in section VII-D, there are two kinds of relationship – the ones that involve human knowledge (dotted lines) and the ones that are automatic in the KDD process (continuous lines). In Figure 3, those relationships show with the same name in order not to overload the diagram. Nevertheless, in the formalization of the ontology they need to be differentiated. For example, the relationship `discovers`, between the classes `Task` and `Pattern`, was formalized in OWL as `discovers_human` and `discovers`, what indicates to the mining tools that will use the ontology that the relationship `discovers_human` involves knowledge and possibly context information, while the relationship `discovers` indicates an activity that can be performed automatically by the mining tool, possibly based on information obtained in previous phases.

In lines 18 to 21, we formalize the relationship `discovers`, quoted in the previous paragraph. In OWL, the relationships among classes are called `Object Property` and they all inherit properties from `owl:topObjectProperty`, as indicated in lines 18 and 21, respectively. `rdfs:domain` and `rdfs:range` (lines 19 and 20) indicate which are the classes involved in a relationship. In the example, relationship `discovers` has classes `Task` as domain and class `Pattern` as range, which means that "Task discovers Pattern" or "the mining task is used to discover patterns". The relationship `discovers_human` is formalized in lines 23 to 26 in a similar way.

In lines 28 to 36 we create the relationships `hasAlgorithm` and `hasAlgorithm_human`, and the domain of those propriedades is class `Task` and its range is class `Algorithm`. Semantically, we casnr ead this relationship as "the mining task has an algorithm".

In lines 38 to 41 the relationship `defines` between classes `Task` and `Technic` is formalized. In this relationship there was no need to represent human participation given that the mining technique will be defined (or automatically suggested by the system) based on the mining task or on the algorithm defined in previous tasks.

As mentioned before, the Meta-DM ontology was formalized in OWL, that uses description logic (DL) to represent knowledge [38]. More information on the syntax and formalism of OWL can be found in [12] and [13].

### *F. Evaluation*

According to METHONTOLOGY methodology [8], adopted for the development of Meta-DM, the evaluation of the ontology consists in performing a technical judgement of it, incluing validation and verification.

Validation assures that the ontology represents the knowledge domain defined in the specification phase. To validate an ontology it is common to use a software system associated with it. In the case of the Meta-DM ontology, the use of an associated software system is deferred to a later date (as described in section VIII). Therefore, only the verification task was performed when evaluating Meta-DM.

Verification refers to the technical process that guarantees the ontology is correct. Such correction can be determined using an inference engine such as Pellet [23], Jena framework [39], FaCT++[40], and others. For the verification of Meta-DM, we used the inference engine Pellet because it is a consolidate tool that is mature in the task of verifying inconsistencies in ontologies. Besides, Pellet can be installed as a plugin in Protégé, is totally compatible with the OWL language and with the SWRL rule definition language (Semantic Web Rule Language) [41].

Using Pellet, we verified if the ontology has any inconsistency. Those inconsistencies could be related to the class disposition (classes in the same hierarchy and disjoint classes), to the relationship among classes (range and domain), to the type of attribute (literal, numeric and others) or to the application

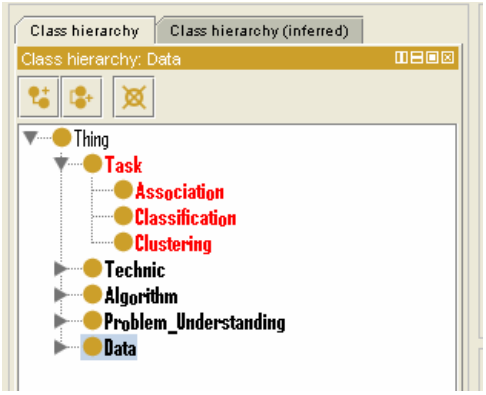


Fig. 5. Classes verified (in black) and not verified (in red) by the inference engine Pellet, used inside the Protégé tool.

of rules (consultations) in the ontology. The fragment of code below shows an inconsistency in in the relationship and in the attribute type.

```
01 :Source rdf:type owl:Class .
02 :path rdf:type owl:DatatypeProperty ;
03 rdfs:domain :Source ;
04 rdfs:range xsd:decimal ;
05 rdfs:subPropertyOf owl:topDataProperty .
06 :path_instance rdf:type :Source ,
07 owl:NamedIndividual ;
08 :path "C:\dados\banco.dbc"^^xsd:decimal .
```

In line 01 the class `Source` is created. In line 02 the property `path` is created. In line 03 we define that the domain of the property `path` is the class `Source` (that is, it is a property of the class `Source`). In line 04 we define the range of the property `path` is the type decimal. Lines 06 to 08 show an instantiation of property `path`, but instead of storing a numerical value we stored a string. As a result, Pellet inference engine accused an inconsistency. When changing the value from string to decimal this inconsistency disappeared.

Figure 5 presents part of the implementation of the Meta-DM ontology classes. The classes in a darker shade were verified with success. Those in a lighter color have not been verified by Pellet yet. When there is an inconsistency, the Pellet inference engine shows a message. Few inconsistencies were detected in Meta-DM and they all were successfully corrected.

### G. Documentation

The methodology used to develop Meta-DM required formal documents for the various phase of the life cycle of the ontology. For Meta-DM we created the following documents:

- Ontology diagram, that is shown in Figure 3;
- Dictionary of the classes, relationships and attributes, with a description of each element in the ontology;
- Description of what reused in the ontology;
- The OWL code generated with ontology formalization;
- Evaluation document.

All these documents can be found in [37].

## VIII. Future Direction

Data mining methodologies supply little detail to the miner how to really perform a step in the process of knowledge discovery in data bases [42]. As stated by Vieira et al. [43], data mining tools widely used in academia, such as Weka[3], focus mainly on data treatment and visualization and do not include instructional and interactive resources on the steps of the KDD process.

In order to help the data mining process a tool called Kira was developed [43]. This tool proposes a series of guidelines to support the user in each step of the data mining process. Theses guidelines abstract a great deal of knowledge needed to perform this process. Kira modules offer facilities to help the user prepare the data, execute mining algorithms and evaluate data results. In the end of each phase, the user is informed on the next step of the process. Kira's goal is to guide the user to perform the KDD steps, even if he or she does not have a detailed knowledge on the data mining process. [43]. The tool was used in experiments with company employees and in classrooms by undergraduate and graduate students.

As stated in section III, our research intends to investigate on ho wto insert human and domain knowledge into data mining tools so that they can work interactively with the miner during the KDD process. In spite of Kira's good results guiding the users in the data mining process, the tool does not consider human intelligence and domain knowledge in the process.

Our future work will focus on adding Meta-DM ontology into an ontologies based architecture to the Kira tool. The goal is to consider the domain knowledge (through an ontology that represents the mining problem domain) and human cooperation in the mining process. The consequences will be (a)Kira'a abilities to guide the user in the data mining process and (b) consider context information to (semi-)automatize some task that are left entirely to the data miner.

## IX. Conclusions

In this paper we present the development of a domain ontology for data mining. The main result present is the Meta-DM ontology, its conceptualization and implementation. Meta-DM intends to supply a common terminology that can be shared and understood by data mining tools. Unlike many other data mining domain ontologies found in literature, Meta-DM identifies and formalizes the phases of data mining where human knowledge must be inserted into the KDD process. This differential is important in order to allow human knowledge to be inserted into mining tools and, consequently, to help or guide the miner during the process of knowledge discovery in databases.

With the ontology implementation in had, our next steps will focus on its integration with the data mining tool Kira [26], in order to improve human participation in this tool. In order to do so, we intend to use Meta-DM to identify and implement more interactive interfaces in Kira.

## References

[1] T. R. Gruber, "A translation approach to portable ontology specifications," *KNOWLEDGE ACQUISITION*, vol. 5, pp. 199–220, 1993.

[2] C.-C. Shen and H.-M. Chuang, "A study on the applications of data mining techniques to enhance customer lifetime value," *WSEAS Trans. Info. Sci. and App.*, vol. 6, pp. 319–328, February 2009. [Online]. Available: http://portal.acm.org/citation.cfm?id=1512690.1512706

[3] H. WITTEN, I and E. FRANK, *Data Mining - Practical Machine Learning Tools and Techniques*, Elsevier, Ed. Second Edition, 2005.

[4] F. M. Pinto and M. F. Santos, "Considering application domain ontologies for data mining," *WSEAS Trans. Info. Sci. and App.*, vol. 6, pp. 1478–1492, September 2009. [Online]. Available: http://portal.acm.org/citation.cfm?id=1718151.1718156

[5] L. Cao, "Domain driven data mining (d3m)," *Data Mining Workshops, International Conference on*, vol. 0, pp. 74–76, 2008.

[6] L. Cao and C. Zhang, "Domain-driven data mining: A practical methodology," *IJDWM*, vol. 2, no. 4, pp. 49–65, 2006.

[7] N. Guarino, "Formal ontology and information systems," *In: Proceedings. Amsterdan: IOS on Formal Ontology and Information Systems (FOIS'98)*, pp. 3 –15, 1998.

[8] M. Fernandez-Lopez, A. Gomez-Perez, and N. Juristo, "Methontology: from ontological art towards ontological engineering," in *Proceedings of the AAAI97 Spring Symposium*, Stanford, USA, March 1997, pp. 33–40.

[9] N. F. Noy and D. L. mcguinness, "Ontology Development 101: A Guide to Creating Your First Ontology," Online, 2001. [Online]. Available: http://www.ksl.stanford.edu/people/dlm/papers/ontology101/ontology101-noy-mcguinness.html

[10] M. Grüninger and M. Fox, "Methodology for the Design and Evaluation of Ontologies," in *IJCAI'95, Workshop on Basic Ontological Issues in Knowledge Sharing, April 13, 1995*, 1995. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.44.8723

[11] M. Uschold and M. King, "Towards a methodology for building ontologies," in *Workshop on Basic Ontological Issues in Knowledge Sharing, held in conjunction with IJCAI-95*, Montreal, Canada, 1995.

[12] W3C, "Owl 2 web ontology language – document overview," W3C Recommendation, October 2009, available at http://www.w3.org/TR/owl2-overview/.

[13] D. L. McGuinness and F. van Harmelen, "Owl web ontology language overview," W3C Recommendation. Disponível em: http://www.w3.org/TR/owl-features/, Fevereiro 2004.

[14] K. S. L. KSL, "Ontolingua," Disponível online http://www.ksl.stanford.edu/software/ontolingua/, 2005, stanford University. Acesso em: 12 jul. 2011.

[15] Loom, "Loom project home page," Disponível online http://www.isi.edu/isd/LOOM/, 2007, university of Southern California – Information Sciences Institute. Acesso em: 12 jul. 2011.

[16] M. Kifer, G. Lausen, and J. Wu, "Logical foundations of object-oriented and frame-based languages," *J. ACM*, vol. 42, pp. 741–843, July 1995. [Online]. Available: http://doi.acm.org/10.1145/210332.210335

[17] E. MANOLA, F.; MILLER, "Rdf primer," *W3C Recommendation*, 2004.

[18] R. V. BRICKLEY, D.; GUHA, "Rdf vocabulary description language 1.0: Rdf schema," *W3C Recommendation*, 2003.

[19] Protégé, "Software protégé," http://protege.stanford.edu/, 2011, stanford University. Acesso em: 12 Jul. 2011.

[20] KAON, "Kaon tool suite," Disponível online: http://kaon.semanticweb.org/, 2011, acesso em: 12 jul. 2011.

[21] SWOOP, "Swoop – semantic web ontology editor," Disponível online: http://code.google.com/p/swoop/, 2011, acesso em: 12 jul. 2011.

[22] M. HORRIDGE, "Owlviz," http://www.co-ode.org/downloads/owlviz/, August 2009, the University Of Manchester.

[23] clark and parseia, "Pellet: Owl 2 reasoner for java," disponível em: http://clarkparsia.com/pellet/, 2 2011, acesso em: 12 jul. 2011.

[24] M. HAN, J.; KAMBER, *Data Mining: Concepts and Techniques*, E. Elsevier, Ed., 2006.

[25] P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer, and R. Wirth, "Cross industry standard process for data mining (crisp-dm 1.0) – step-by-step data mining guide," http://www.crisp-dm.org/CRISPWP-0800.pdf, 2000.

[26] E. F. MENDES, "Kira: Uma ferramenta instrucional para apoiar a aplicação do processo de mineração de dados," Master's thesis, Faculdade de Ciências Exatas e da Natureza, da Universidade Metodista de Piracicaba – UNIMEP, 2009.

[27] R. RAKOTOMALALA, "Tanagra: a free software for research and academic purposes," http://eric.univ-lyon2.fr/ ricco/tanagra/en/tanagra.htm, 2005.

[28] Oracle, "Oracle data mining," Disponível em http://www.oracle.com/us/products/database/options/data-mining/index.html, 2011.

[29] S. Sharma and K.-M. Osei-Bryson, "Organization-ontology based framework for implementing the business understanding phase of data mining projects," *Hawaii International Conference on System Sciences*, pp. 1–10, 2008.

[30] L. Zheng and X. Li, "An ontology reasoning architecture for data mining knowledge management," *Wuhan University Journal of Natural Sciences*, vol. 13, pp. 396–400, 2008, 10.1007/s11859-008-0403-y. [Online]. Available: http://dx.doi.org/10.1007/s11859-008-0403-y

[31] P. Brezany, I. Janciak, and A. M. Tjoa, "Ontology-based construction of grid data mining workflows," in *Data Mining with Ontologies: Implementations, Findings, and Frameworks*. Hershey, 2007. [Online]. Available: http://eprints.cs.univie.ac.at/472/

[32] J. Vanschoren and L. Soldatova, "Exposé: An ontology for data mining experiments," in *International Workshop on Third Generation Data Mining: Towards Service-oriented Knowledge Discovery (SoKD-2010),*, Sep. 2010, pp. 31–46. [Online]. Available: https://lirias.kuleuven.be/handle/123456789/273222

[33] P. Panov, S. Dzeroski, and L. Soldatova, "Ontodm: An ontology of data mining," *Data Mining Workshops, International Conference on*, vol. 0, pp. 752–760, 2008.

[34] P. Panov, L. N. Soldatova, and S. Dzeroski, "Towards an ontology of data mining investigations." in *12th International Conference on Discovery Science (DS'09)*, ser. Lecture Notes in Computer Science, J. Gama, V. S. Costa, A. M. Jorge, and P. Brazdil, Eds., vol. 5808. Springer, 2009, pp. 257–271. [Online]. Available: http://dblp.uni-trier.de/db/conf/dis/dis2009.html#PanovSD09

[35] P. Panov, S. Džeroski, and L. N. Soldatova, *Inductive Databases and Constraint-Based Data Mining*. Springer, 2010, ch. Representing Entities in the OntoDM Data Mining Ontology, pp. 27–58.

[36] D. Beckett and T. Berners-Lee, "Turtle – terse rdf triple language," W3C Team Submission. Disponível em: http://www.w3.org/TeamSubmission/turtle/, Janeiro 2008.

[37] E. A. Yokome, "Uma ontologia para inserir conhecimento humano em ferramentas de mineração de dados," Master's thesis, Faculdade de Ciências Exatas e da Natureza, da Universidade Metodista de Piracicaba – UNIMEP, 2011.

[38] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed. Prentice-Hall, Englewood Cliffs, NJ, 2003.

[39] Jena, "Jena – a semantic web framework for java," Disponível online: http://jena.sourceforge.net/. Acesso em 26 de julho de 2011., 2011.

[40] FaCT++, "Fact++ owl-dl reasoner," Disponível online: http://owl.man.ac.uk/factplusplus/. Acesso em 26 de julho de 2011., 2011.

[41] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean, "Swrl: A semantic web rule language combining owl and ruleml," W3C Member Submission, May 2004, disponível em http://www.w3.org/Submission/SWRL/. Acesso em 26 de julho de 2011.

[42] M. Charest, S. Delisle, O. Cervantes, and Y. Shen, "Intelligent data mining assistance via cbr and ontologies," in *Proceedings of the 17th International Conference on Database and Expert Systems Applications (DEXA'06)*, 2006, pp. 1–5.

[43] M. T. P. Vieira, A. E. A. Silva, C. Peixoto, E. F. Mendes, and R. S. Gomide, "Kira – a tool based on guides and domain knowledge to instruct data mining," in *Proceedings of the IADIS International Conference Applied Computing*, vol. 2. Rome, Italy: IADIS, 2009, pp. 12–16.