OntoDM-KDD: Ontology for Representing the Knowledge Discovery Process

Panče Panov[1], Larisa Soldatova[2], and Sašo Džeroski[1]

[1] Jožef Stefan Institute, Jamova cesta 39, Ljubljana, Slovenia
{pance.panov,saso.dzeroski}@ijs.si
[2] Brunel University, London, United Kingdom
larisa.soldatova@brunel.ac.uk

Abstract. In this article, we present an ontology for representing the knowledge discovery (KD) process based on the CRISP-DM process model (OntoDM-KDD). OntoDM-KDD defines the most essential entities for describing data mining investigations in the context of KD in a two-layered ontological structure. The ontology is aligned and reuses state-of-the-art resources for representing scientific investigations, such as Information Artifact Ontology (IAO) and Ontology of Biomedical Investigations (OBI). It provides a taxonomy of KD specific actions, processes and specifications of inputs and outputs. OntoDM-KDD supports the annotation of DM investigations in application domains. The ontology has been thoroughly assessed following the best practices in ontology engineering, is fully interoperable with many domain resources and easily extensible. OntoDM-KDD is available at `http://www.ontodm.com`.

Keywords: Knowledge Discovery in Databases, CRISP-DM, Data Mining Investigation, Data Mining, Domain Ontology.

1 Introduction

Recent surveys of research challenges for knowledge discovery in databases (KDD) and data mining (DM) list the mining of different types of structured data in a uniform fashion, the use of domain knowledge, and the support for complex KDD processes as the top-most open issues in the domain [1–3]. Much of the research in recent years has also focused on the automation and overall support of the KDD process. This involves development of standards for performing the KDD process as well as formal representations of the processes in the form of workflows [4, 8, 10, 13]. Specific issues addressed include methods that automate the composition of data mining operations into executable workflows. Finally, providing a mechanism for recording of results and the experimental settings of the DM experiments obtained by executing the workflows on sets of data is becoming important for ensuring the repeatability and reuse of results [14].

One of the most prominent proposals for standardizing the process of knowledge discovery in the context of representing and performing data mining investigations is the Cross Industry Standard Process for Data Mining (CRISP-DM) [4].

J. Fürnkranz, E. Hüllermeier, and T. Higuchi (Eds.): DS 2013, LNAI 8140, pp. 126–140, 2013.

It is a process model that describes data mining investigations performed in practical applications. The CRISP-DM process model is based on commonly used approaches that expert data miners use to tackle and solve the practical problems in the domain.

The formalization of scientific investigations has also proved to be a prominent area of research. It includes providing a formal representation of objects and processes involved in scientific investigations in a knowledge representation framework, such as terminologies, taxonomies, and ontologies. The largest developments in this sense have taken place in biological and biomedical domains (e.g., the Robot Scientist project[5]). In addition, the state-of-the-art also includes initiatives to support and unify the representational mechanisms for recording scientific investigations under a single framework (e.g., the Open Biomedical Ontologies (OBO) Foundry [6]).

In the domain of KDD and DM, there exist several proposals of domain ontologies but the majority of them are light-weight application oriented ontologies aimed at covering a particular use-case in data mining. Initial systems that include ontologies are used to systematically describe the processes in machine learning and DM (e.g., the IDA system [7]). Next, there are ontology developments aimed to support workflow composition and planing of workflows [7–10], support of data mining applications on the GRID [11, 12], and support of meta-learning and meta-mining [13]. Finally, there are ontologies designed to support machine learning experiments in the context of experiment databases [14].

In our previous work [15, 16], we formally represented and described the complex domain of data mining by developing OntoDM, a general-purpose domain ontology of DM that takes into account the state-of-the-art developments in the area of formalization of scientific investigations. In this paper, we present the OntoDM-KDD ontology, a novel sub-ontology module of OntoDM. OntoDM-KDD (v.1) introduces the data mining investigations as a representational mechanism to describe the complete process of KDD, based on the CRISP-DM process model. The ontology includes a taxonomy of KD specific processes, actions and representations of inputs and outputs. Finally, the ontology has been thoroughly assessed following the best practices in ontology engineering, evaluated and validated by the applications on a use case.

2 Design

The OntoDM-KDD ontology is based on the CRISP-DM process model [4]. Its main goal is to be general enough to allow the representation of knowledge discovery processes and data mining investigations performed in practical applications. Based on this main goal, we identified a list of competency questions that our ontology is designed to answer.

Table 1. Examples of OntoDM-KDD competency questions

What is the description of a DM investigation X?
What is the set of publications about the investigation for a DM investigation X?
What is the DM investigation that is reported by a publication X?
What is the set of actions realized by the KD phase X for a DM investigation Y?
What is the set of DM investigations that realize an action X in a KD phase Y?
What is the process that precedes process X in KD phase Y for a DM investigation Z?

Examples of OntoDM-KDD competency questions are listed in Table 1. From the list of questions, we can see that the ontology need to include basic information entities for representing data mining investigations, such as action specifications, reports, and textual entities. Furthermore, the ontology need to contain processual entities that are parts of the knowledge discovery process.

In order to ensure the interoperability of OntoDM-KDD with other resources, the OntoDM-KDD ontology follows the OBO Foundry design principles [1]. These include, for example, the use of an upper level ontology, the use of formal ontological relations, single inheritance, and the re-use of already existing ontological resources where possible [6]. The use of these design principles enables cross-domain reasoning, facilitates wide reusability of the developed ontology, and avoids duplication of ontology development efforts.

OntoDM-KDD imports the upper level classes from the Basic Formal Ontology (BFO1.1)[2] and formal relations from the OBO Relational Ontology (RO)[3] [17] and uses an extended set of RO relations. BFO an RO were chosen as they are widely accepted, especially in the biomedical domain. Following best practices in ontology development, the OntoDM-KDD ontology reuses appropriate classes from a set of ontologies, that act as mid-level ontologies. These include the Ontology for Biomedical Investigations (OBI)[4], the Information Artifact Ontology (IAO)[5], and the Software Ontology (SWO)[6]. Classes that are referenced and reused in OntoDM-KDD are imported by using the Minimum Information to Reference an External Ontology Term (MIREOT) principle [18] and extracted using the OntoFox web service[7].

OntoDM-KDD is expressed in OWL-DL[8], a de facto standard for representing ontologies. The ontology is being developed using the Protege[9] ontology editor. The ontology is freely available at `http://www.ontodm.com` and at Bio Portal[10].

[1] OBO Foundry: `http://obofoundry.org/crit.shtml`
[2] BFO: `http://www.ifomis.org/bfo`
[3] RO: `http://purl.org/obo/owl/OBO_REL`
[4] OBI: `http://obi-ontology.org/page/Main_Page`
[5] IAO: `http://code.google.com/p/information-artifact-ontology`
[6] SWO: `http://theswo.sourceforge.net`
[7] OntoFox: `http://ontofox.hegroup.org`
[8] OWL-DL: `http://www.w3.org/TR/owl-guide`
[9] Protege: `http://protege.stanford.edu`
[10] BioPortal: `http://bioportal.bioontology.org`

3 The Structure of OntoDM-KDD

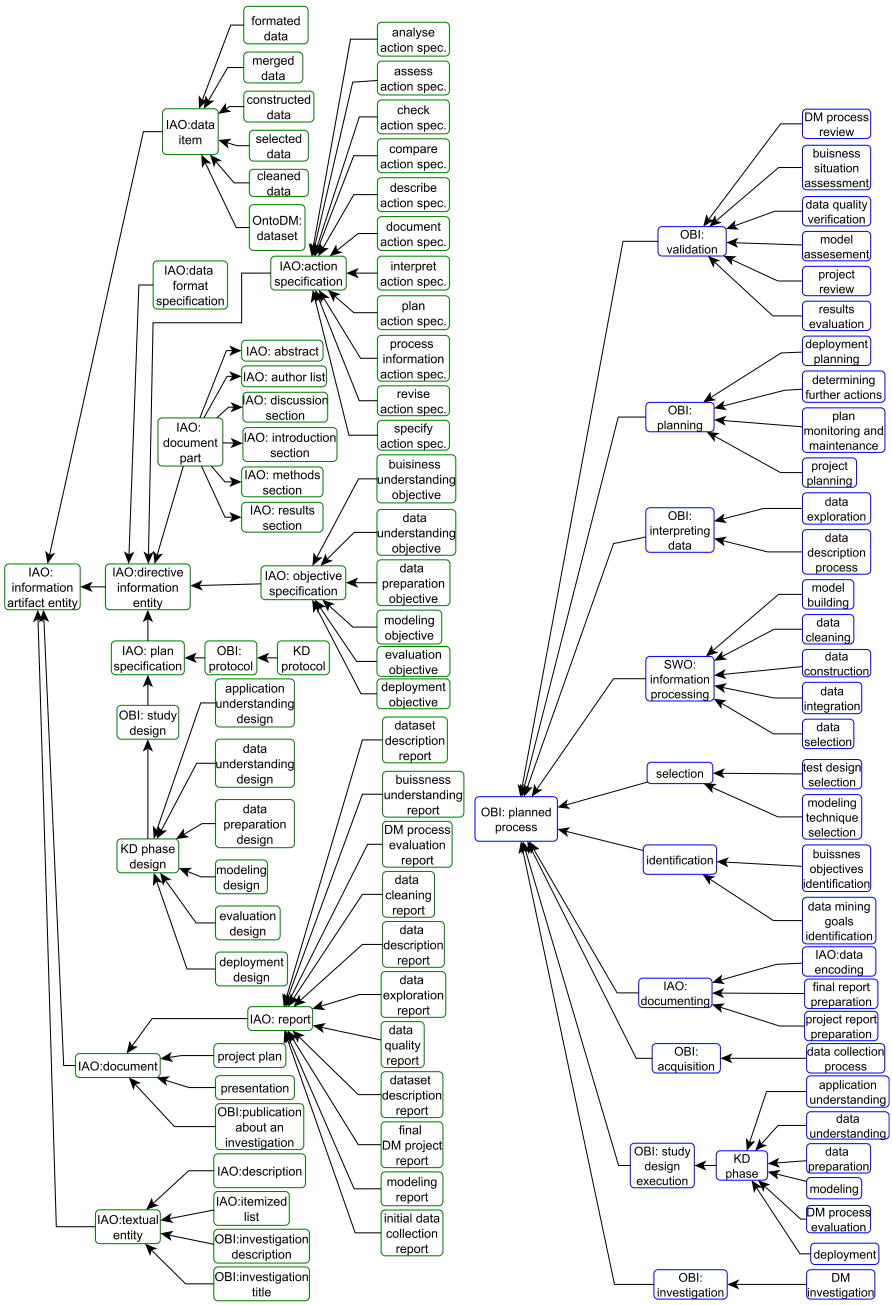
The CRISP-DM process model, at the top level, is organized into six phases: business understanding phase, data understanding phase, data preparation phase, modeling phase, evaluation phase, and deployment phase [4]. It defines the outputs of each CRISP-DM phase and the second-level generic tasks. For example, the data understanding phase consists of four generic tasks: collect initial data, describe data, explore data, and verify data quality. The level of specialized tasks (third level) describes how the generic tasks should be carried out in specific situations, in terms of activities. For example, the describe data task includes activities for volumetric analysis of data, assessment of the attribute types and values, etc. The fourth level, the level of process instances, describes the actions, decisions and results of an actual data mining investigation performed in the domain of interest.

For the purpose of representing data mining investigations, it is very important to have the ability to represent entities that deal with information, such as data, documents, reports, models, algorithms, protocols, etc. We thus incorporate and further extend some classes of the IAO ontology. The IAO ontology is a mid-level ontology describing information content entities (e.g., documents), processes that consume or produce information content entities (e.g., documenting), material bearers of information (e.g., journals), and relations in which one of the relata is an information content entity (e.g., is-about).

Another important representational aspect is representation of processes. In OntoDM-KDD, we use and further extend classes from the OBI ontology, such as the OBI process taxonomy, which includes general processes such as documenting, planing, validation, etc. The OBI ontology aims to provide a standard for the representation of biological and biomedical investigations. It supports consistent annotation of biomedical investigations regardless of the particular field of study and is fully compliant with the existing formalisms in biomedical domains [19]. In addition, OBI defines an investigation as a process with several parts, including the planning of an overall study design, executing the designed study, and documenting the results. Finally, in OntoDM-KDD we include the SWO class *Information Processing* that represents processes in which input information is analysed or transformed in order to produce an output information.

In OntoDM-KDD we distinguish two description layers based on the mid-level ontologies that it extends (Fig.1). The first layer is the specification layer, that deals with information entities needed to describe and represent the DM investigations. The second layer is the application layer that deals with processual entities in order to represent processes that occur in a DM investigation.

The specification layer (Fig. 1(a)) consists of classes that are extensions of the IAO class *Information Content Entity*. At the top level, it includes classes such as *Data Item*, *Directive Information Entity*, *Document*, *Document Part* and *Textual Entity*. The *Directive Information Entity* class is further extended with *Action Specification*, *Data Format Specification*, *Objective Specification*, and *Plan Specification*. In addition, we also reuse the *Study Design* and *Protocol* classes from the OBI ontology.



(a) Specification layer (b) Application layer

Fig. 1. The structure of the OntoDM-KDD ontology. The imported classes include in their name the source ontology label (IAO, OBI, SWO, OntoDM) as a prefix, while the native OntoDM-KDD classes are shown without such a label.

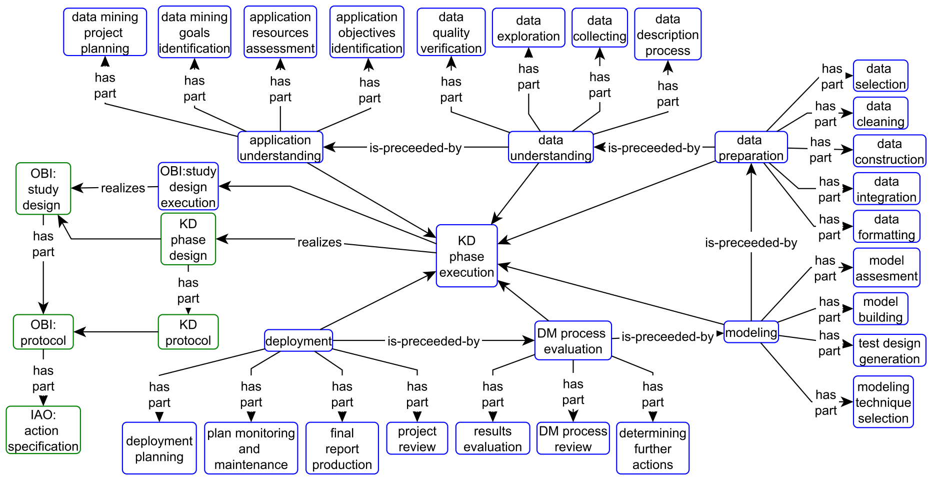


Fig. 2. KD phase execution and the *has-part* taxonomy of KD specific processes. The unlabeled arrows represent *is-a* relations. Classes represented with green boxes belong to the specification layer, while the blue boxes belong to the application layer.

The application layer (Fig. 1(b)) consists of classes that are extensions of the OBI class *Planned Process*. These include general classes of processes such as: *Validation*, *Planning*, *Interpreting Data*, *Information Processing*, *Selection*, *Identification*, *Documenting*, and *Acquisition*. Finally, the application layer includes the OBI *Investigation* class, which we further extend to define and represent a *Data Mining Investigation*.

4 Mapping the CRISP-DM Model to OntoDM-KDD

The phases level from the CRISP-DM model is represented in OntoDM-KDD with two aspects. In the specification layer, we represent the specification of the phases with the *KD Phase Design* class (Fig 2). It is a subclass of OBI *Study Design* which is comprised of *Protocols*. For example, the *Data Understanding Design* contains the specification of the data understanding phase.

The *KD Phase Design* is realized during *KD Phase Execution*. *KD Phase Execution* is represented as a processual entity in the application layer and is extended with KD specific phases from the CRISP-DM process model (Fig 2). These include *Application Understanding*, *Data Understanding*, *Data Preparation*, *Modeling*, *DM Process Evaluation*, and *Deployment* class. For example, *Data Preparation* process is a *KD Phase Execution* and realizes the *Data Preparation Design*. In this version of OntoDM-KDD, we can represent a sequential ordering of the KD phases by using the *is-preceded-by* relation (Fig.2).

Similar as the phases, the generic tasks from each phase from the CRISP-DM model are represented in OntoDM-KDD with two aspects. In the specification layer, we present specification of the tasks with the *KD Protocol Class*. For example, the *Data Understanding Design* contains as parts the *Data Collecting Protocol*, the *Data Describing Protocol*, the *Data Exploring Protocol* and the *Data Quality Verification Protocol*.

The executions of the protocols are represented in the application layer, and are parts of the *KD Plase Execution* process. For example, *Data Understanding* contains as parts sub-processes: *Data Collecting*, *Data Exploration*, *Data Description Process*, and *Data Quality Verification*. Each of the sub-processes is a sub-class of a more general processes class. For example, *Data Collecting* is a sub-class of *Aquisition* process.

The activities from the specialized tasks level of the CRISP-DM model are represented in OntoDM-KDD as actions. One of the most important parts of the specification layer is the taxonomy of KDD specific actions, represented by the extension of the *Action Specification* class. The action specification defines the actions that are realized in the processes. At the first level, we have the more general actions such as *Analyze Action*, *Assess Action*, *Check Action*, *Compare Action*, *Describe Action*, *Document Action*, *Interpret Action*, *Plan Action*, *Process Information Action*, *Revise Action*, and *Specify Action*. At the second level, the general actions are extended with KDD specific actions. Finally, each *KD Protocol* contains a set of action specifications as parts. For example, the *Data Exploring Protocol* includes the *Explore Data Action* and *Formulate Hypothesis Action*.

An *Investigation* is a planned process and includes the *Planning*, *Documenting*, and *Study Design Execution* processes (Fig. 3). Furthermore, an investigation is described with an *Investigation Title*, *Investigation Description*, and *Investigation Identifier*. In addition, the investigation produces as output a *Conclusion Textual Entity*. Finally, a *Publication About an Investigation* is a document about it and it is an output by the documenting sub-process. OntoDM-KDD defines a *DM Investigation* class as an extension of the OBI *Investigation* class (Fig. 3). A *DM Investigation* has as its part the *KD Phase Execution* process.

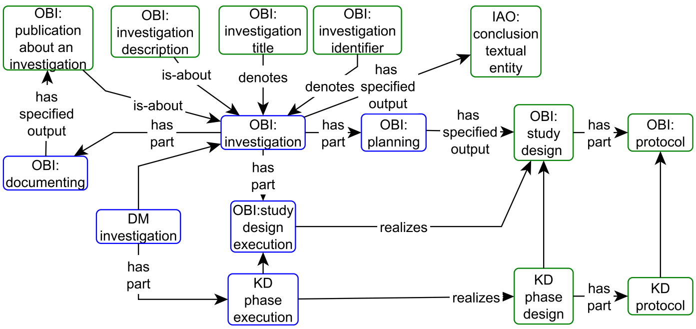


Fig. 3. The data mining investigation class in OntoDM-KDD

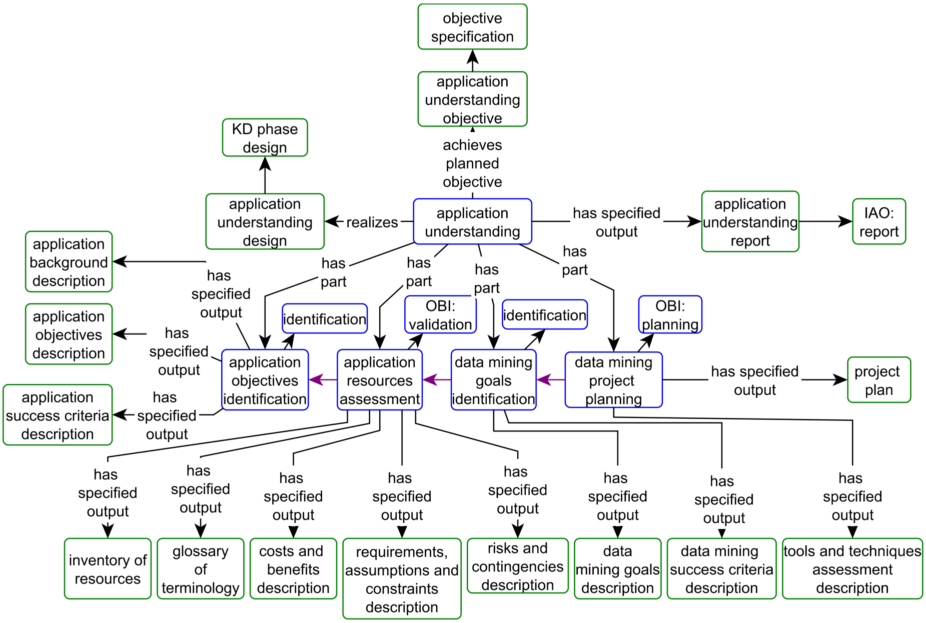


Fig. 4. Application understanding process in OntoDM-KDD. The unlabeled arrows represent *is-a* relations, while coloured arrows represent *is-preceded-by* relation.

5 Example: Application Understanding Process

In this section, we present an example of representation of one phase from CRISP-DM in OntoDM-KDD. The initial phase in a DM investigation focuses on identifying the objectives and requirements of the investigation, from an application (or business) perspective. In CRISP-DM, this phase was named business understanding, while in OntoDM-KDD we generalize it as application understanding. The goal is to convert the knowledge about the application domain into a data mining problem definition and to generate a plan for achieving the application objectives.

The *Application Understanding* class is a sub-class of *KD Phase* and represents a planned process (Fig. 4). In the ontological vocabulary, the *Application Understanding* process can be defined as a *KD Phase* that realizes an *Application Understanding Design*, achieves the planned objective an *Application Understanding Objective* and has specified output *Application Understanding Report*. The *Application Understanding Process* includes as parts four sub-processes: *Application Objectives Identification*, *Application Resources Assessment*, *Identification of Data Mining Goals* and *Generation of a Project Plan*.

The process of *Application Objectives Identification* is a sub-class of the more general class of *Identification* processes. In this process, a data analyst (active participant or agent) needs to identify in detail, from the application (or business) perspective, what are the objectives to be achieved by applying DM to

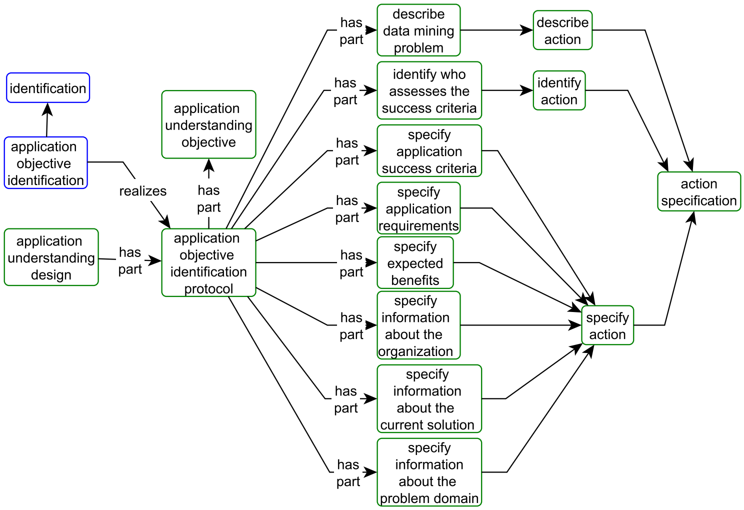


Fig. 5. The process of objective identification in OntoDM-KDD

the application domain at hand. At the end of the process, the analyst needs to produce as output an *Application Background Description*, an *Application Objectives Description*, and an *Application Success Criteria Description*.

The process of *Application Objective Identification* is a realization of an *Application Objective Identification Protocol* (Fig. 5). The protocol, which is a part of the *Application Understanding Design*, contains a specification of the actions that are realized in the process, such as *Describe Data Mining Problem*, *Specify Application Success Criteria*, *Identify who Assesses the Success Criteria* and others. These action specifications are subclasses of general classes of actions, such as *Describe Action*, *Specify Action* and *Identify Action*. In the OntoDM-KDD, we represent and provide action specifications for all processes.

The process of *Application Resources Assessment* is a sub-class of the OBI *Validation* process. It involves assessing the information about all resources, constraints, assumptions and other factors that need to be considered in order to determine the data mining goals and the project plan. The outputs of this process include: an *Inventory of Resources*, a *Glossary of Terminology*, *Costs and Benefits Description*, a *Requirements Assumptions and Constraints Description*, and a *Risks and Contingencies Description*.

The process of *Identification of Data Mining Goals* is a sub-class of the *Identification* process. The objective of this process is to produce a specification of the data mining goals and establish a set of data mining success criteria which can be used to evaluate the success of the data mining investigation at hand. The output of this process includes a *Data Mining Goals Description* and a *Data Mining Success Criteria Description*.

The process of *Data Mining Project Planning* is a sub-class of the OBI *Planning* process. The objective of this process is to produce a specification of a plan in order to achieve the data mining and application goals. The outputs of the process include a *Project Plan* and a *Tools and Techniques Assessment Description*.

6 Evaluation

We assess the quality of OntoDM-KDD from three different evaluation aspects. First, we analyze a set of ontology metrics. Then, we assess how well the ontology meets a set of predefined design criteria and ontology best practices. Finally, we assess how the ontology meets a set of predefined competency questions.

A variety of ontology metrics is available for assessing ontologies. We use the statistical ontology metrics from the Protégé software and the BioPortal web service (Tab. 2(a)). OntoDM-KDD has 264 classes, 34 relations and 2091 axioms. The size of the ontology is comparable with the average size of the OBO Foundry ontologies and the complexity (the number of relations and axioms) is higher than the average.

Table 2. OntoDM-KDD Evaluation

(a) Statistical metrics

Axiom count	2091
Class count	264
Individual count	0
DL expresivity	SHI
SubClassOf axiom count	521
DisjointClasses axiom count	53
Relations count	34
Annotation axioms count	1178

(b) An example of a competency question formalized in SPARQL-DL

What actions are realized by the KD phase X for the investigation Y?
Q(act):-Type(?act,action_specification), PropertyValue(?prot,has-part,?act), Type(?prot,protocol) PropertyValue(?kddphdesign,has-part,?prot), Type(?kddphdesign,kdd_phase_design), PropertyValue(?kddphplan, is-concretization-of, ?kddphdesign), PropertyValue(?x,realizes,?kddphplan), Type(x,kdd_phase_execution), PropertyValue(y,has-part,x), Type(y,investigation).

The ontology has been constructed following the best in ontology engineering and design criteria. The set of design principles (in total 29) is divided into four groups: scope and structural assessment; naming and vocabulary assessment; documentation and collaboration assessment; and availability, maintenance, and use assessment. The results of the evaluation are summarized on the ontology web page (`www.ontodm.com`). In sum, the design principles were closely followed during the development of OntoDM-KDD.

Following the recommendations by Gruniger and Fox [20], we first defined the ontology's requirements in the form of competency questions that the ontology must be able to answer (see above). Furthermore, having defined the language of the ontology, the competency questions are defined formally as an entailment with respect to the axioms in the ontology. In this way, one can evaluate the ontology and claim that it is adequate if the questions can be formulated in the language of the ontology. For that purpose, we formulated the questions using SPARQL-DL query language[11] [21] for querying OWL-DL ontologies. SPARQL-DL is a subset of the SPARQL language[12]. In Tab. 2(b), we show an example of an OntoDM-KDD competency question formulated in SPARQL-DL.

7 Usecase: Annotation of Data Mining Investigations

In this section, we present an example of how OntoDM-KDD can be used to annotate DM investigations in application domains. For this purpose, we focus on a DM investigation titled "Estimating forest properties from remotely sensed data using DM", published in a journal article by Stojanova et al. [22].

The DM investigation aimed at modeling forest properties, such as vegetation height and canopy cover, from remotely sensed data, by using DM algorithms. The final goal of this investigation was to use the models of the properties to generate forest maps that can be deployed in forest management and forest decision support systems. The DM investigation included: the study of the application domain; collection of data; preparation of the data; modeling of the forest properties; evaluation of the modeling process and deciding on the best model; generation of the forest property maps; and finally deployment of the generated maps in forest management systems.

In Fig. 6, we present a part of the annotation of this investigation. First, we define *dm investigation*[13] as an instance of the *DM Investigation* class that has as parts instances of the *planning*, *documenting*, and a *kd phase execution* processes (or their child classes). In addition, *'Estimating forest properties from remotely sensed data' denotes* the investigation and represents its title and the *investigation description* instance that *is-about* the investigation.

The documentation process *has-specified-output* a *Publication About An Investigation*, which represents an entity that *is-about* an investigation. An instance of this class is used to represent the journal article. In addition, this instance has as parts document part instances, such as *abstract*, *author list*, *institution list*, *introduction to a publication about an investigation*, *methods section*, *results section*, *discussion section of a publication about an investigation*, *conclusion to a publication about an investigation*, and *references section*. Finally, *'Estimating vegetation height and canopy cover from remotely sensed data with machine learning' denotes* the publication's title.

[11] SPARQL-DL: `www.w3.org/2001/sw/wiki/SPARQL-DL`
[12] SPARQL: `http://www.w3.org/TR/rdf-sparql-query/`
[13] Notation: with non-capitalized italics we denote instances of classes.

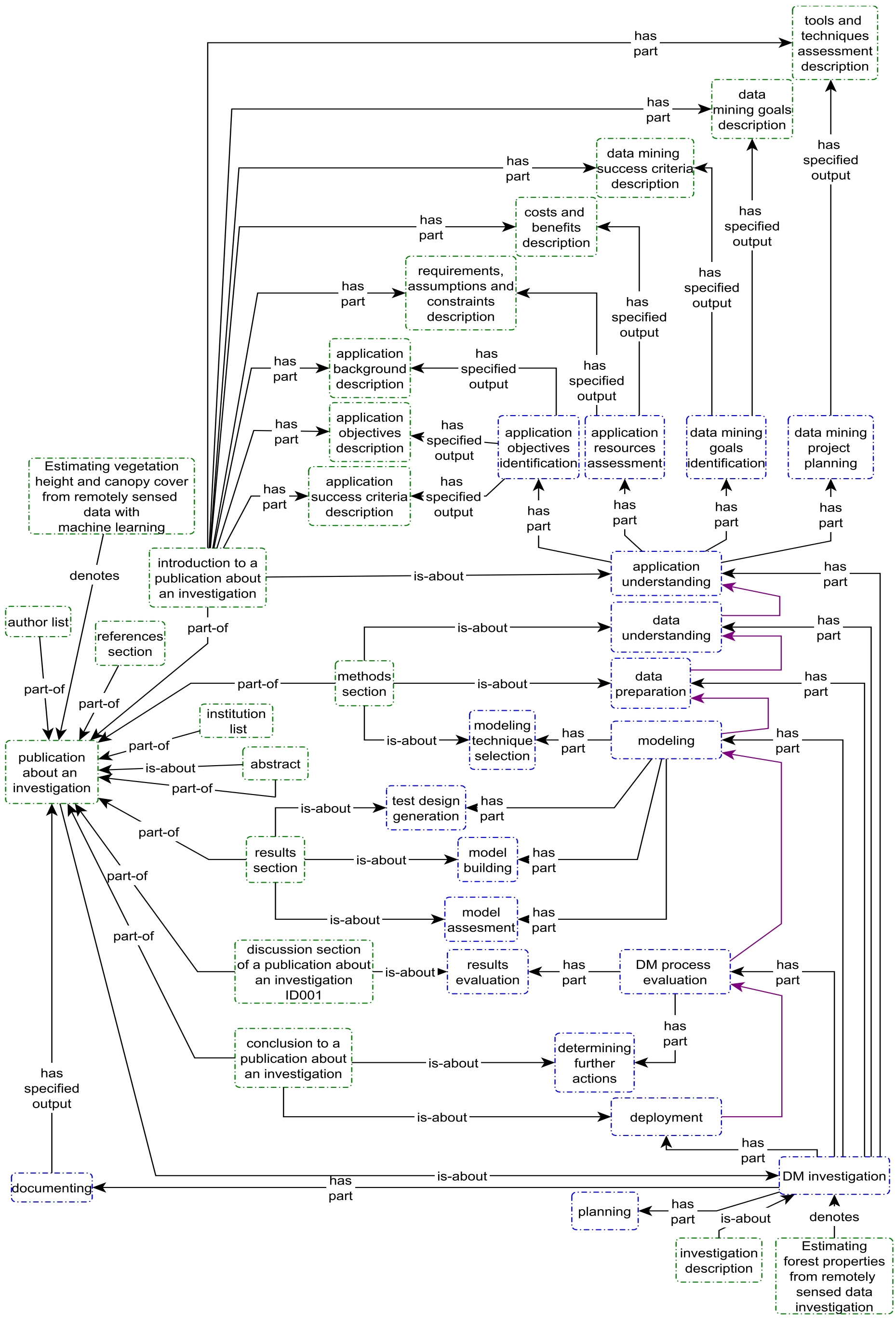


Fig. 6. Part of an annotation of a DM investigation summarized in a journal article with terms from the OntoDM ontology

The introduction part of the article, represented by the *introduction to a publication about an investigation* instance, *is-about* the application understanding process. It has as parts descriptions that are outputs of processes that compose the application understanding process. For example, the introduction includes description instances such as *application background description*, *application objectives description*, and *application success criteria description*. These descriptions are instances of the *Textual Entity* class and are outputs of the *application objective identification* process instance. The same holds also for the other process instances that compose the application understanding process.

The methods section, represented by the *methods section* instance, contains parts that are about the data understanding process, the data preparation process and modeling technique selection (as a sub-process of modeling). More specifically, it contains description of the study area (the Kras region), data sources (LiDAR and Landsat), data descriptions (descriptive and output variables), and description of the DM techniques to be used. These descriptions are outputs of the the sub-processes of data understanding and data preparation, and the modeling technique selection process [14].

The results section, represented by the *results section* instance, contains parts that are about test design generation process, the model building process, and the model assessment process. These are all sub-processes of modeling. More specifically, it contains descriptions of the experimental design, DM algorithms applied, evaluation procedure and results (best models in terms of predictive performance and maps of vegetation height and canopy cover for the best model). These descriptions are outputs of the sub-processes of test design generation, model building and model assessment.

The discussion section, represented by the *discussion section of a publication about an investigation* instance, contains parts that are about a results evaluation process, a sub-process of a DM process evaluation. More specifically, it contains descriptions of a comparison of the performances of all applied DM techniques, a comparison to previous work, and a discussion of the produced maps of vegetation properties.

The conclusion section, represented by the *conclusion to a publication about an investigation* instance, contains parts that are about the deployment process and determining further actions process, a sub-process of DM process evaluation. More specifically, it contains a summary of contributions, a description of the deployment of the produced maps, and a description of envisioned future work.

The considered example demonstrates that OntoDM-KDD has the expressivity to annotate the key concepts pertinent to typical DM investigations. OntoDM-KDD annotations would facilitate machine amenable recording of the information about how DM investigations have been carried out, enable accurate comparison of such investigations and reasoning e.g. about what DM methods work better for what applications. OntoDM-KDD is also an important resource to facilitate text mining of DM relevant literature.

[14] For simplicity/readability reasons, for all other parts Fig. 6 contains only the upper level processes.

8 Conclusion and Future Work

In this paper we proposed OntoDM-KDD, an ontology for representing the knowledge discovery based on the CRISP-DM process model. The OntoDM-KDD ontology was designed and implemented by following ontology best practices and design principles. It used an upper-level ontology BFO as a template, included formally defined relations, and reused classes from other ontologies for representing scientific investigations.

The ontology introduced a two-layered representation mechanism and provided a taxonomy of KD specific processes and actions. In addition, it provided a specification of inputs and outputs of the KD specific processes. Furthermore, the ontology introduced the data mining investigation entity as representational mechanism for describing and annotating data mining investigations in application domains (e.g., biology, forestry, etc). The OntoDM-KDD ontology has been applied for annotation of data mining investigations summarized in journal articles. In addition, the SWO ontology version 0.4 reused some of the OntoDM-KDD classes for representing data pre-processing and modeling processes.

In the context of representation of the complete knowledge discovery process, most of the current ontologies focus only on representing the modeling phase. Some of the ontologies, such as DMOP [13] and Expose [14], also provide entities that cover the data preparation phase of a KDD process, but do not provide ontological support for the complete KDD process. The strength of the OntoDM-KDD ontology is that it provides support for representation of the complete knowledge discovery process from application understanding to deployment.

In future developments of the OntoDM-KDD ontology, we plan to align the ontology to the BFO 2.0 top level ontology that is in final phases of preparations. Furthermore, we plan to apply the ontology for representation of data mining investigations in different application domains of data mining.

References

1. Yang, Q., Wu, X.: 10 challenging problems in data mining research. International Journal of Information Technologies and Decision Making 5(4), 597–604 (2006)
2. Kriegel, H.P., et al.: Future trends in data mining. Data Mining and Knowledge Discovery 15, 87–97 (2007)
3. Dietterich, T., Domingos, P., Getoor, L., Muggleton, S., Tadepalli, P.: Structured machine learning: the next ten years. Machine Learning 73 (2008)
4. Chapman, P., Kerber, R., Clinton, J., Khabaza, T., Reinartz, T., Wirth, R.: The CRISP-DM process model. Discussion Paper (1999)
5. King, R., et al.: The Automation of Science. Science 324(5923), 85–89 (2009)
6. Smith, B., et al.: The OBO foundry: coordinated evolution of ontologies to support biomedical data integration. Nat. Biotech. 25(11), 1251–1255 (2007)
7. Bernstein, A., Provost, F., Hill, S.: Toward intelligent assistance for a data mining process: An ontology-based approach for cost-sensitive classification. IEEE Trans. on Knowl. and Data Eng. 17(4), 503–518 (2005)
8. Žáková, M., Kremen, P., Železný, F., Lavrač, N.: Automating knowledge discovery workflow composition through ontology-based planning. IEEE Transactions on Automation Science and Engineering 8(2), 253–264 (2010)

9. Diamantini, C., Potena, D.: Semantic annotation and services for KDD tools sharing and reuse. In: ICDMW 2008: Proceedings of the 2008 IEEE International Conference on Data Mining Workshops, pp. 761–770. IEEE Computer Society (2008)
10. Kietz, J., Serban, F., Bernstein, A., Fischer, S.: Towards cooperative planning of data mining workflows. In: Proceedings of Workshop on Third Generation Data Mining: Towards Service-Oriented Knowledge Discovery, pp. 1–13 (2009)
11. Cannataro, M., Comito, C.: A data mining ontology for GRID programming. In: Proc. of Wshp. on Semantics in Peer-to-Peer and Grid Computing, pp. 113–134 (2003)
12. Brezany, P., Janciak, I., Tjoa, A.M.: Ontology-based construction of grid data mining workflows. In: Data Mining with Ontologies: Implementations, Findings and Frameworks, pp. 182–210. IGI Global (2007)
13. Hilario, M., et al.: A data mining ontology for algorithm selection and Meta-Mining. In: Proceedings of Workshop on Third Generation Data Mining: Towards Service-Oriented Knowledge Discovery, pp. 76–88 (2009)
14. Vanschoren, J., Blockeel, H., Pfahringer, B., Holmes, G.: Experiment databases - a new way to share, organize and learn from experiments. Machine Learning 87(2), 127–158 (2012)
15. Panov, P., Džeroski, S., Soldatova, L.N.: OntoDM: An ontology of data mining. In: ICDMW 2008: Proceedings of the 2008 IEEE International Conference on Data Mining Workshops, pp. 752–760. IEEE Computer Society (2008)
16. Panov, P., Soldatova, L., Džeroski, S.: Representing entities in the OntoDM data mining ontology. In: Inductive Databases and Constraint-Based Data Mining, pp. 27–58. Springer, New York (2010)
17. Smith, B., et al.: Relations in biomedical ontologies. Genome Biology 6(5), R46 (2005)
18. Courtot, M., et al.: MIREOT: The minimum information to reference an external ontology term. Applied Ontology 6(1), 23–33 (2011)
19. Brinkman, R.R., et al.: Modeling biomedical experimental processes with obi. Journal of Biomedical Semantics 1(suppl. 1), S7 (2010)
20. Grüninger, M., Fox, M.: Methodology for the Design and Evaluation of Ontologies. In: IJCAI 1995, Workshop on Basic Ontological Issues in Knowledge Sharing (April 13, 1995)
21. Sirin, E., Parsia, B.: SPARQL-DL: Sparql query for OWL-DL. In: 3rd OWL Experiences and Directions Workshop (OWLED 2007) (2007)
22. Stojanova, D., Panov, P., Gjorgjioski, V., Kobler, A., Dzeroski, S.: Estimating vegetation height and canopy cover from remotely sensed data with machine learning. Ecological Informatics 5(4), 256–266 (2010)