XML Schema and Topic Map Ontology for Background Knowledge in Data Mining

Tomáš Kliegr[1], Vojtěch Svátek[1], Milan Šimůnek[1], Daniel Šťastný, Andrej Hazucha

University of Economics, Prague, Dept. Information and Knowledge Engineering, Nám. Winstona Churchilla 4, 130 67 Praha 3, Czech Republic, {tomas.kliegr,svatek,simunek,xstad17,xhaza00}@vse.cz

Abstract. Background (or sometimes referred to as domain) knowledge is extensively used in data mining for data pre-processing and for nugget-oriented data mining tasks: it is essential for constraining the search space and pruning the results. Despite the costs of eliciting background knowledge from domain experts, there has been so far little effort to devise a common exchange standard for its representation. This paper proposes the Background Knowledge Exchange Format (BKEF), a lightweight XML Schema for storing information on features and patterns, and the Background Knowledge Ontology (BKOn), as its semantic abstraction. The purpose of BKOn is to allow reasoning over and integration of analysed data with existing domain ontologies. We show an elicitation interface producing BKEF and discuss the possibilities for integration of such background knowledge with domain ontologies.

1 Introduction

Elicitation of knowledge from experts has long been known as a crucial research topic in the field of expert systems, and its importance is now starting to rise in data mining applications, too. Background (or sometimes referred to as domain) knowledge is extensively used in preprocessing of data for most mining algorithms. It has special importance in association rule mining, where it is used to separate the nuggets from rules conveying uninteresting information.

Despite the potential of expert-provided background knowledge for improving the quality of data mining results, there has been so far little research effort onselecting pieces of information that should be collected and little standardization efforts on devising a common format for representation of background knowledge. This paper presents one of the first attempts to address these problems by introducing the Background Knowledge Exchange Format (BKEF) XML Schema. Simultaneously, to allow reasoning and integration of analysed data with existing domain ontologies, we propose a semantic abstraction over BKEF – the Background Knowledge Ontology (BKOn).

This paper is organized as follows. Section 2 gives an account of the proposed design objectives of a background knowledge specification. Section 3 introduces

its elementary building blocks and section 4 gives account of specificalities for association rules. The proposed BK specification consisting of BKEF XML Schema and the BKOn ontology is described in Sections 5 and 6 respectively. The new possibilities that BKEF and BKOn open in the areas of automating data mining tasks and result postprocessing are sketched in Section 7. The conclusion presents an outlook for future work.

2 Design Objectives

The work presented here reacts to the pressing need for an industry standard that would provide a common way of conveying pieces of background knowledge that express expertise related to features and patterns relevant to datasets in a given domain. Hence, although in the common case the knowledge acquistion is driven by the need for knowledge pertaining to a specific mining task and specific dataset, the standard should impose such principles that would foster reuse of the knowledge in a different task-dataset scenario. While the work presented here has experimental character, it follows some of the design guidelines that, we believe, should be addressed by any serious attempt on an industry standard specification.

We will use the term *background knowledge producer* to denote a computer program, such as a specialized elicitation interface, used by the domain expert to input his/her background knowledge related to the data mining task.

The *background knowledge consumer*, in turn, denotes a computer program that uses background knowledge (BK). We consider the following types of BK consumers: data preprocessing algorithms, data mining algorithms, postprocessing algorithms and semantic knowledge bases.

2.1 One size does not fit all

The standard should be constituted by an XML Schema and an ontology to accommodate for the different needs of background knowledge producers and consumers.

It may seem natural that the language in which the specification is defined is selected so that its expressivity is at least such as required by the most demanding consumer type, which is the semantic knowledge base. The semantic knowledge base [11] interlinks mining models, background knowledge and domain ontologies, and as such it would take advantage of background knowledge comming directly in a semantic format such as RDF/OWL [2] or the Topic Maps' XTM [7]. However, there are reasons for not using a semantic format as the primary standard used by data mining and knowledge elicitation software. The main ones include:

- poor readability due to structural complexity
- verbosity
- the need for specialized, not widely available APIs

Therefore, we propose using an XML Schema as an interchange format between background knowledge consumers and background knowledge producers. To foster the interoperability on the semantic level, the specification should also define a semantic version of the XML Schema (an ontology) and a transformation between the schema and the ontology. This transformation is to be executed on the side of the BK consumer.

2.2 Background Knowledge Consumer Requirements

The primary goal of the specification is to provide pieces of information that can be automatically processed by background knowledge consumers and doing so can enhance their functioning.

#	Consumer Type	Information	Utilization
1	Data Preprocessing	Similar value grouping	Decreasing the granularity
2	Data Mining	Search space constraints	Localizing the search
3	Postprocessing	Known patterns	Pruning
4	Semantic KBs	Annotations	Search

Table 1. Frequent use cases for background knowledge

An overview of requirements on the specification posed by the individual consumers is given in Table 1. This table was constructed based on the analysis of requirements of the LISp-Miner mining suite[1] and the SEWEBAR framework[2] as Semantic KB for association rules, but the authors conjecture that the table should be, with some changes, applicable to other mining tasks and algorithms.

Requirements on storing the types of information of types 1–3 require inherently no semantics and can be met by the XML Schema specification. Since indisputably one of the consumers of background knowledge is the human data analyst, the specification should also provide the domain expert with the possibility to complement the machine-readable values with a free-text annotation.

The requirements of the Semantic KB consumer type are addressed in subsection 2.3. While closely linked to background knowledge and essential for the Semantic KB, machine-readable annotations fall out of the scope of the background knowledge specification.

2.3 Integration with Other Specifications

The background knowledge specification discussed here has strong links with PMML, the widely adopted standard for data mining model interchange [3]. The

[1] `http://lispminer.vse.cz`
[2] `http://sewebar.vse.cz/`
[3] `http://dmg.org`

proposed specification plays the same role for background knowledge as PMML does for mining models. For background knowledge consumers to be able to apply this knowledge together with knowledge gained from PMML, the need for alignment with PMML arises.

While one of the key design objectives is independence of the BK specification of a specific dataset/task scenario, the bond between the BK specification and a concrete dataset or mining model should be established in a separate mapping specification. Further, we briefly introduce an attempt for such a specification dubbed FML (Field Mapping Language).

PMML is backed by an XML Schema, which eases the design of the mapping. A more complex problem arises with the requirements imposed by the Semantic KB consumer type. The purpose of Semantic KBs is to perform reasoning, integration and search over the data. From this arises the necessity to annotate the entities that emerged during the background knowledge elicitation process (such as features, values and patterns) with an association to relevant concepts in other ontologies or with unstructured sources. Since this annotation information transcedes the scope of a single dataset, we suggest to support it with a standalone specification (an XML Schema or an ontology) so that it is not a direct part of BKEF, but is only linked with it. Since the only BKEF consumer in our framework that has direct use for this kind of information is the Semantic KB, a semantic format such as RDF/OWL could be more convenient for storing the annotations than XML Schema. Additionaly, this annotation can aid the process of automatic mapping of BKEF onto a specific dataset resulting into an FML specification.

3 Basic Concepts

3.1 Metaattribute

The basic building block of a background knowledge specification is a *metaattribute* [14], which is an abstraction representing the underlying property of a data-field. There is a hierarchical structure between metaattributes. The metaattribute on the finest granularity level is referred to as *atomic metaattribute*. Other attributes are called *group metaattributes*.

Since a property can be sometimes measured in different ways, most commonly using different units, each metaattribute has multiple *formats*. Actually, most pieces of information relating to a metaattribute are format-dependent. Specifically, a format can contain:

- *a value range*,
- *standard value binning(s)*,
- *a collation*.

Since the specification is intended to be used in conjunction with a dataset, where a datafield always conforms to one metaattribute format, it is advantageous to introduce a common term *Meta-field* for an atomic metaattribute-format pair.

Similarly *Meta-field Value* is an abstraction of a possible 'value' of a metafield – value or interval falling within the scope given in the value range or one of the groupings.

3.2 Patterns

Known relationships between metaattributes are captured using *patterns*. Since often the pattern only applies to a specific format or involves a value, the notion of meta-field and meta-field value is central for their definition.

The purpose of patterns is to be used in conjunction with the data mining algorithm, most commonly either in the algorithm itself or in the further processing of results. As such, it is difficult to introduce a unified framework for pattern representation that would be equally usable for all types of data mining tasks and algorithms. Therefore the specification should propose suitable types of patterns for the main data mining algorithms (such as classification, clustering or association rule mining).

4 Background Knowledge for Association Rule Mining

We introduce two types of patterns that were designed to aid the association mining algorithms; their prospective utilization for other types of mining algorithms is a matter for further research. These two types are *Mutual Influences* and *Background Knowledge Association Rules*.

A Background Association Rule (BAR) has the form of

$$\kappa \ \approx_{[\iota]} \ \lambda \ [/\chi] \tag{1}$$

.

Here the Antecedent κ, Consequent λ and Condition χ are Boolean Meta-attributes and $\approx$ is a type of 4ft-quantifier. The optional ι explicitly corresponds to value(s) of Interest Measures associated with the 4ft-quantifier. The BAR is Conditional if the Condition χ is present.

4ft-quantifier corresponds to a set of conditions (*interest measures*) defined on the four-field contingency table, which is a quadruple of natural numbers $\langle a, b, c, d\rangle$ so that: a is the number of objects(rows) from the data matrix satisfying φ and ψ, b satisfying φ and $\neg\psi$, c satisfying $\neg\varphi$ and ψ and d the number of objects satisfying $\neg\varphi$ and $\neg\psi$. A *Boolean Meta-attribute* is a recursive structure comprising conjunctions, disjunctions and negations of combinations of individual items (Metafield-Value pairs). A Boolean Meta-attribute is *Basic* or *Derived*. A *Basic Boolean Meta-Attribute* has the form of $b(\sigma)$, where the *Coefficient* σ is a subset of possible Values of Meta-Field b. A *Derived Boolean Attribute* is a conjunction or disjunction of Boolean Meta-attributes, or a negation of a Boolean Meta-attribute.

The Background Association Rule can be input independently into the Pattern component of a BKEF document, or as an Atomic Consequences element

within a Mutual Influences element. The notion of *Mutual Influence* comes out of research by Rauch & Šimůnek [14], who proposed to use it as a knowledge elicitation aid.

5 Background Knowledge Exchange Format

The Background Knowledge Exchange Format (BKEF) is defined by an XML Schema and used for storing mining models of a particular knowledge domain. The BKEF XML Schema consists of two main building blocks: definitions of **meta-attributes** and definitions of patterns. A metaattribute is understood as an abstraction of the ultimate property of the mining model [14] with all characteristics explained so far, hence metaattributes are simultaneously comprised in the BKEF XML Schema. Mutual influences among the metaattributes together form a *pattern*. A simplified schema is shown in Fig. 1.

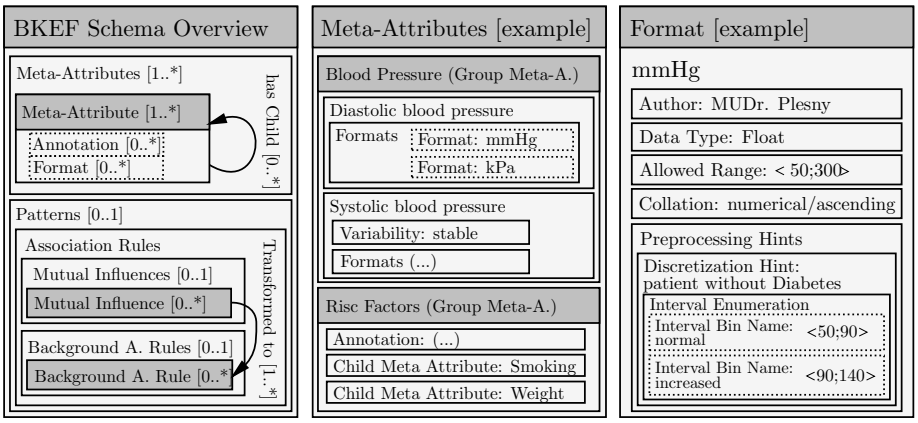

Fig. 1. Schema of BKEF

5.1 Metaattribute Definitions in BKEF

The XML Schema restricts meta-attributes to a two-level hierarchy. The base level encompasses indivisible `MetaAttributes`[4] (level = 0) - basic layer, evenly *atomic metaattribute*. The upper level comprises groups of the `MetaAttribute` elements (level = 1); each group contains an unlimited number of the `Meta-Attribute`.

[4] *Typewriter* text labels on particular elements of the BKEF XML Schema where it is necessary to refer about XML elements for the proper understanding.

Groups of meta-attributes A general collection of `MetaAttribute` elements. The group should have a name, unique identification and at least one link to the `MetaAttribute` of level = 0 (which is called `ChildMetaAttribute` from this point of view).

Meta-Attribute The main focus of the `MetaAttribute` is the multiple definition of the `Format` as the property could be expressed in different ways of measurement. The `Annotation` together with the author's name are used for additional information on different authors. See an example:

```
<Annotation>
<Text>Measured in 2009</Text>
  <Author>MUDr. Plesny</Author>
</Annotation>
```

The `Variability` of the `MetaAttribute` is expressed either as *stable* or *actionable* whereas the unchangeable properties in the mining model are stable. E.g. the date of birth cannot be changed, thus this metaattribute is referred to as *stable*. If we for example expect that the systolic blood pressure can be influenced by some other property, we refer to the `Variability` as actionable [17], otherwise it can also be a stable `MetaAttribute`; this depends on the mining model and its research targets. An atomic `MetaAttribute` element contains at least one `Format`.

Format The `Format` is identified by a unique name (within the collection) and encompasses the following elements: `Author`, `Annotations` (which is a collection of particular annotations), `DateType`, `ValueType`, `ValueAnnotations`, `AllowedRange`, `Collation`, `PreprocessingHints` and `ValueDescriptions`.

Each `Annotation` consists of the name of an author and the commentary - each format could be commented through the `Annotations` (collection of `Annotation` elements). The `Author` of the `Format` is self-explanatory, as a value of the `DataType` is used some of the common data type readable by the intended consumer BK (string, integer, boolean etc.). The `ValueType` content distinguishes between cardinal, nominal, ordinal and a real number. Commonly used are values as nominal and ordinal for qualitative meta-attribute and cardinal (which means an interval or a rational number) for quantitative metaattributes [13].

The `ValueAnnotations` element is defined for the commentary to particular values: each value can be commented separately more than once. The particular annotation has the same format as the `Annotation`.

The `AllowedRange` element denotes a value boundary of the particular format of the `MetaAttribute`. Thus the formats of the same values can differ. The range can be defined by `Interval` for quantitative values (maximum and minimum) or by `Enumeration` for qualitative values. See an example of allowed range defined by an interval:

```
<Interval>
<LeftBound type="closed" value="2"/>
<RightBound type="closed" value="15"/>
</Interval>
```

The `Collation` expresses a commonly accepted arrangement of the *greater than* relation between format values, if such an arrangement exists. This is essential for interpretation of the be *greater than* relationship between values [14]. The BKEF XML Schema differentiates between easily sortable numerical values and qualitative values whose sequence is expressed by the enumeration as depicted on the following example:

```
<Collation type="Numerical" sense="Ascending" />
```

respectively

```
<Collation type="Enumeration" sense="Ascending">
  <Value>elementary</Value>
  <Value>secondary</Value>
<Value>university</Value>
</Collation>
```

The `PreprocessingHints` element conveys to a BK Consumer the information on how to prepare data. The current version of the BKEF XML Schema allows one or more `DiscretizationHint` elements as the only possible child elements of the `Preprocessing Hint`. The values of the `DiscretizationHint` are assorted into discreet counterparts. There can be more than one preprocessing hint, for example depending on the desired granularity of the metaattribute values. The way of discretization is set up by `ExhaustiveEnumeration` or `IntervalEnumeration`. It reflects all intended values of the metaattribute designated for the BK consumer and consecutive mining tasks. The element `IntervalEnumeration` is used for numerical values, as seen from an example:

```
 <IntervalEnumeration>
  <IntervalBin name="normal">
   <Annotation>...</Annotation>
<Interval>
<LeftBound type="closed" value="60"/>
<RightBound type="closed" value="88"/>
</Interval>
</IntervalBin>
  <IntervalBin name="overweight indicator">
   <Annotation>...</Annotation>
<Interval>
<LeftBound type="closed" value="88"/>
<RightBound type="closed" value="140"/>
</Interval>
```

```
  </IntervalBin>
</IntervalEnumeration>
```

An example of `ExhaustiveEnumeration` for non-numerical values is:

```
<ExhaustiveEnumeration>
<Bin name="yes">
   <Annotation>...</Annotation>
   <Value>yes</Value>
  </Bin>
  <Bin name="no">
   <Annotation>...</Annotation>
   <Value>no</Value>
  </Bin>
</ExhaustiveEnumeration>
```

The exhaustive enumeration corresponds with the Map Values (where the values are defined as a table) of PMML 3.2 [4].
There are another two variations of interval enumeration: `Equifrequent` (the number of intervals is given and the interval boundaries are determined automatically so that the frequency of values falling into each interval is roughly identical) and `Equidistant` (given exact lenght of an interval). The *Discretization Hint* element does not include the value sets aggregation (known from PMML[4]), otherwise the clear and expressive discretization hint structure is one of the strengths of the BKEF XML Schema.

The `Value Descriptions` element is used for characteristics of particular values. It uses the `Interval` or `Value` elements for numerical and non-numerical values, respectively.

```
<ValueDescriptions>
<ValueDescription type="Significant">
   <Annotation>...</Annotation>
    <Interval>
      <LeftBound type="closed" value="100"/>
      <RightBound type="closed" value="150"/>
    </Interval>
</ValueDescription>
</ValueDescriptions>
```

In general, setting of the `Collation`, `PreprocessingHints` and `ValueDescriptions` is not a question of an exact method, as their determination is fully dependent on the domain expert and a particular mining task.

5.2 Patterns in BKEF

The current BKEF XML Schema allows to define `MutualInfluences`, which are a base for the BAR.

A `MutualInfluences` contains at least one `MutualInfluence`, which forms a relation between two metaattributes $A \rightarrow B$.

```
<Influence type="Positive-bool-growth" id="20" arity="2">
<KnowledgeValidity>Unknown</KnowledgeValidity>
<MetaAttribute role="A" name="weight">
<RestrictedTo><Format name="kg"/></RestrictedTo>
</MetaAttribute>
<MetaAttribute role="B" name="Hyperlipoproteinemy">
<RestrictedTo>
<Format name="boolean value">
<Value format="boolean value">yes</Value>
</Format>
</RestrictedTo>
</MetaAttribute>
</Influence>
```

`KnowledgeValidity` can have two values – *Unknown*, *Proven* or *Rejected* – regarding the mining task result. The metaattribute appearing in the influence might be restricted to the `Format` or even particular value (which should be linked with the corresponding `Format` of the atomic `MetaAttribute`).

6 Background Knowledge Ontology

The *Background Knowledge Ontology* is a semantic abstraction of the BKEF XML Schema introduced in section 5. The purpose of the BKEF XML Schema is to rigidly enumerate what types of background knowledge are acceptable and in what format. To this, BKOn adds information on relations between the pieces of background knowledge by explicitly linking them through typed associations, thus adding machine-readable semantics for background knowledge consumers. The most prominent consumer is the Semantic KB, which utilizes these relations for reasoning.

Adding semantics to the BKOn results in reshuffling of the BKEF content. The design guidelines that were followed when translating BKEF nodes to BKOn ontology topics are the same that were followed when creating the Association Rule Mining Ontology from PMML as described in [10]. Reenumerating the guidelines is out of the scope of this paper, nevertheless the main principle is simple – allow for automatic transformation of BKEF XML documents into instances of the ontology concepts while making the resulting ontology as clean as possible.

To achieve this, the following prominent changes in BKOn compared to BKEF were made

- some concepts that were only implicitly present in the BKEF XML Schema are explicitly present in BKOn,

– some BKEF XML nodes do not have a corresponding concept in the ontology as they are contained in the newly created concepts,
– explicit superclasses for closely related topics are introduced.

Some of the concrete examples of these changes are as follows: *Metafield* becomes an explicit ontology concept and a concept directly corresponding to the `Format` BKEF element is no longer explicitly present in the ontology. One instance of the `Metafield` concept is created from each pair of `Format` element and its containing `Metaattribute` element.

The *Metafield Binned Content* is used as a superclass for `EnumerationBin` and `IntervalBin`, and *Metafield Raw Content* as a superclass for `Interval` and `Value`. Both these newly introduced concepts have the *Metafield Content* superclass.

We make a reference transformation implemented as an XSLT stylesheet available[5]. The gist of BKOn is depicted on Figure 2.

7 Exploiting BKEF and BKOn in the Data Mining Loop

This section demonstrates a possible use case of BKEF and BKOn, in conjunction with the academic data mining system LISp-Miner and the SEWEBAR framework. LISp-Miner is an academic system for KDD developed at University of Economics, Prague [1] for teaching and research in the area of KDD. It consists of several procedures covering the entire process of KDD as described in the CRISP-DM methodology.[6]. The SEWEBAR (for: Semantic Web – Analytical Reports) framework involves a content management system and a semantic knowledge base for creating and sharing knowledge relating to data mining tasks. It is based on the Joomla! CMS and the Ontopia Topic-Map-based Knowledge Base.[7]

This section goes through elicitation of background knowledge within SEWEBAR-CMS, its linking with the mined data using the FML, using it to localize search and prune results within the LISp-Miner system, and finally through its semantic postprocessing, again in SEWEBAR-SKB. The description of the workflow is illustrated in a data mining task whose purpose is to find novel knowledge in a cardiological dataset.

7.1 Background Knowledge Elicitation

The first implementation of background knowledge elicitation was integrated into the LM KnowledgeSource and LM DataSource modules [19] of the LISp-Miner system. However, it emerged later that it is more suitable for domain experts to use a web-based system. This prompted the development of the BKEF Editor (see [5]), as one of the modules of SEWEBAR-CMS.

[5] At `http://sewebar.vse.cz`
[6] `www.crisp-dm.org`
[7] See `ontopia.net` and `joomla.org` for more info

Example Starting the aforementioned data mining use case, consider a medical expert, a cardiologist, who initiates the data mining process. The cardiologist uses the BKEF editor to convey her knowledge of the characteristics that are recorded about cardiological patients and indicates known and interesting relationships appearing in these characteristics.

7.2 Linking Background Knowledge with Mined Data

The main challenge faced is how to properly match data fields that are used in the current data mining task with the semantically equivalent metaattributes. This problem can be divided into two steps: choosing the right BKEF file for the domain being mined and matching metaattributes and their values with data fields and data field values. While this problem is a unique one, it bears significant resemblance with problems that are addressed in ontology alignment and schema mapping research [6]. Since fully automated construction of a reliable mapping seems to be unfeasible given the state of the art in ontology matching and schema mapping, a semi-automated mapping approach is proposed. There is an ongoing work on a web-based system that would propose such a mapping based on a mixture of schema mapping and ontology alignment techniques, which would then have the user confirm the proposed mappings. The result of this mapping is a Field Mapping Language (FML) document. The data mining system will use a web service to locate and retrieve correct FML and BKEF files.

Example The data analyst working with the cardiological dataset searches for BKEF files related to the dataset. Two such files are found. The first one is a BKEF file created by the cardiologist; the second is from a different domain, but it contains general medical fields such as Age or Blood pressure. Once the metaattributes are mapped to datafields though the semiautomatic process highlighted above, the data mining software can use the Preprocessing hints associated with mapped metaattributes to automatically perform discretization and outlier treatment.

7.3 Background Knowledge for Localizing Search

In LISp-Miner, the first implemented use of background knowledge was to guide users in the process of defining Local Analytical Questions (LAQs). That is to properly define what kind of patterns in the analyzed data we are looking for. LAQs are based on pre-defined patterns that lead to different types of questions asked and therefore to different data mining procedures used for answering them. LAQs were first proposed in [18].

Based on actual background knowledge the first type of LAQ pattern could be to mine for yet unknown influences between two groups of attributes (e.g. social status attributes and health status attributes). Or, another LAQ pattern could be used to pinpoint some condition under which some relationship stored into ontology does not hold (e.g. Concerning men above 50 living in Prague it IS

NOT TRUE that...”). Solving such a LAQ could lead to updates of background knowledge.

Example The data analyst is looking for guides to help him/her design the parameters of the data mining task. Based on the information contained in the BKEF pattern section, the data mining system shows that it is already known by the experts that high waist-hip ratio is associated with hypertension. Based on this piece of information, the data analyst instructs the system to look for exceptions to this rule – i.e. to find subsets of data (circumstances) where the high waist-hip ratio is NOT associated with hypertension.

7.4 Background Knowledge for Result Pruning

Another prospective use of background knowledge is pruning of the results of data mining that are of no value for experts (e.g. of patients giving birth to child, at least 99 % are women). If such a relationship is stored in BKEF, no implicational[8] association rule with the attribute concerning ability to give birth to a child on the left side (antecedent) and gender on the right side (succedent) will be placed into results.

Even more useful is pruning in case of a function-like dependency between two attributes, e.g. Age and Height. In general, there is a clear dependency between the age of people and their height. When described by association rules many specific rules will emerge in results, which is undesirable. Instead, a better-suited procedure of the KL-Miner (see e.g. [16] could be (automatically) used and many association rules related to this dependency could be pruned from the results and represented by a single KxL-fold contingency table to describe this function like dependency as a single pattern.

Example The cardiologist is not interest in obvious facts in the results. So all patterns expressing already known relationship between the high waist-hip ratio and hypertension are automatically pruned from the results (if not explicitly overruled by the data analyst). This covers all the derived patterns, i.e. even pruning of extended patterns that logically follow from the simple implication of the form waist-hip ratio(high) =¿ hypertension(true).

7.5 Background Knowledge for Postprocessing

SEWEBAR-CMS [11] accepts mining models in PMML sent through a web service by the data mining system. The BKEF XML files are already present in the system as they originate there. Combining these pieces of information, the analyst conveys the results to the domain expert through a textual analytical report using special report-authoring tools within the CMS [20]. PMML and BKEF documents are semantized according to the Data Mining Ontology [10]

[8] A subclass of association rules [12].

and the BKOn ontology. They are interlinked and stored in the SEWEBAR-SKB, which answers queries issued from the CMS. The queries are issued in the tolog query language, which is a combination of Prolog and SQL. The results of the queries are returned by the Semantic KB in XML, using an XSLT transformation converted to HTML and returned to the user.

Example To communicate the results to medical specialists, the data analyst creates a textual analytical report summarizing his/her findings. In the report s/he also includes the semantic query against the Semantic KB for related association rules that were found in previous tasks, including those executed over different datasets.

8 Conclusions

The main purpose of this paper was to discuss the requirements on a standard for exchange of background knowledge in data mining. The paper also details an attempt for such a specification consisting of the BKEF Schema and BKOn ontology. Practical experience with these formats has already been described in [11], including the interlinking of BKOn with a data mining ontology for association rules introduced in [10] and examples of semantic queries over the merged ontologies.

Future work will primarily address the issue of 'smart' interlinking to domain ontologies, presumably using ontology patterns[9]. This will allow to explicitly disambiguate vague notions, e.g. that of hypertension, which can equally be a summarization of several measurements or a permanent characteristic of a patient. In relation to that, a version of BKOn based on the RDF/OWL formalism (in addition to the Topic Map one) will be built.

9 Acknowledgment

This work has been partly supported from grant no IGA 15/2010 of UEP and by grant GAR 201/08/0802 of Czech Grant Agency.

References

1. LISp-Miner: academic system for KDD [online]. [cit. 2010-03-20], available from WWW: ¡http://lispminer.vse.cz¿
2. OWL Web Ontology Language Overview. W3C Recommendation, 10 February 2004. `http://www.w3.org/TR/owl-features/`
3. W3C: XSL Transformation. Online: `www.w3.org/TR/xslt`. 1999
4. DMG: PMML 3.2 Specification, Online: `http://www.dmg.org/pmml-v3-2.html`
5. Balhar, J., Kliegr, T., Stastny D., Vojir S.: Elicitation of Background Knowledge for Data Mining. In: Znalosti 2010, Czech Republic, February 2010.

[9] `http://www.ontologydesignpatterns.org`

6. Euzenat J. and Shvaiko P.: *Ontology matching*. Springer-Verlag. 2007. ISBN 3-540-49611-4.
7. Garshol L. M., Moore G.: Topic Maps i?1 XML Syntax. ISO/IEC JTC1/SC34, `http://www.isotopicmaps.org/sam/sam-xtm/`.
8. Garshol, L.M.: TMRAP -i?1 Topic Maps Remote Access Protocol. In: Maicher, L., Sigel, A., Garshol, L.M. (eds.) TMRA 2006. LNCS (LNAI), vol. 4438. Springer, Heidelberg (2007)
9. Garshol, L.M.: Towards a Methodology for Developing Topic Maps Ontologies. In: Maicher, L., Sigel, A., Garshol, L.M. (eds.) TMRA 2006. LNCS (LNAI), vol. 4438. Springer, Heidelberg (2007)
10. Kliegr, T., Ovecka M., Zemanek, J.: Topic Maps for Association Rule Mining. In: Proc. TMRA 2009. University of Leipzig 2009.
11. Kliegr M., Ralbovský M., Svátek, V, Šimůnek M., Jirkovský V., Nemrava J., Zemánek J.: Semantic Analytical Reports: A Framework for Post-Processing Data Mining Results. In: Foundations of Intelligent Systems (ISMIS'09). Springer Verlag, LNCS, 2009, 88i?198.
12. Rauch, J.: Classes of Association Rules: An Overview. In: Studies In Computational Intelligence. Springer 2008.
13. Rauch J.: Considerations on Logical Calculi for Dealing with Knowledge in Data Mining. In: Advances in Data Management. Studies in Computational Intelligence, Volume 223/2009, Springer 2009.
14. Rauch J., Šimůnek M.: Dealing with Background Knowledge in the SEWEBAR Project. In: Knowledge Discovery Enhanced with Semantic and Social Information. Studies in Computational Intelligence, Volume 220/2009, Springer 2009.
15. Rauch J., Šimůnek M.: Alternative Approach to Mining Association Rules. In Lin T Y, Ohsuga S, Liau C J, and Tsumoto S (eds): Data Mining: Foundations, Methods, and Applications, Springer-Verlag, 2005.
16. Rauch, J., Šimůnek, M., Lín, V.: Mining for Patterns Based on Contingency Tables by KL-Miner First Experience. In: Foundations and Novel Approaches in Data Mining. Berlin : Springer-Verlag, 2005, s. 155167. ISBN 3-540-28315-3. ISSN 1860-949X.
17. Rauch, J., Šimůnek, M.: Action Rules and the GUHA Method: Preliminary Considerations and Results. ISMIS 2009: 76-87
18. Rauch, J., Šimůnek, M.: LAREDAM Considerations on System of Local Analytical Reports from Data Mining. Toronto 20.05.2008 – 23.05.2008. In: Foundations of Intelligent Systems. Berlin : Springer-Verlag, 2008, pp. 143–149.
19. Šimůnek, M.: Academic KDD Project LISp-Miner. In: Advances in Soft Computing - Intelligent Systems Desing and Applications. Heidelberg : Springer-Verlag, 2003, s. 263272. ISBN 3-540-40426-0.
20. Vojir S.: SEWEBAR - gInclude - Analytical Report Design using gInclude. In: Znalosti 2010, Czech Republic, in Czech, February 2010.

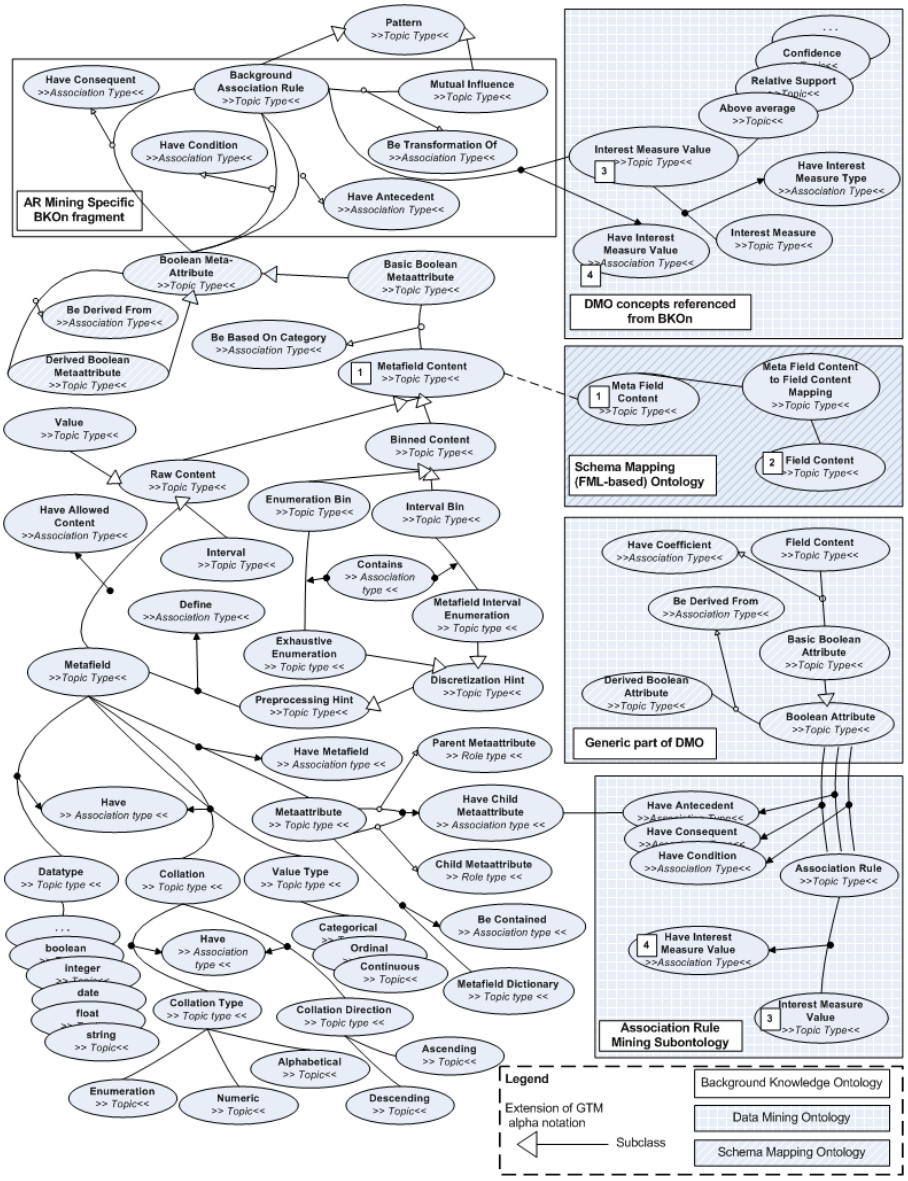

Fig. 2. Background Knowledge Ontology Overview