JMLR: Workshop and Conference Proceedings 10: 55-65 The Fourth Workshop on Feature Selection in Data Mining

Feature Extraction for Machine Learning: Logic–Probabilistic Approach

Vladimir Gorodetsky GOR@IIAS.SPB.SU
Prof. of Computer Science,
Chief Scientist of Practical Reasoning, Inc. and The Intelligent Systems
Laboratory of The St. Petersburg Institute
for Informatics and Automation of the Russian
Academy of Science.
SPIIRAS, 39, 14-th Line V.O.,
St. Petersburg, 199178, Russia

Vladimir Samoylov SAMOVL@IIAS.SPB.SU
research fellow of Practical Reasoning, Inc. and The Intelligent Systems
Laboratory of The St. Petersburg Institute
for Informatics and Automation of the Russian
Academy of Science.
SPIIRAS, 39, 14-th Line V.O.,
St. Petersburg, 199178, Russia

Editor: Huan Liu, Hiroshi Motoda, Rudy Setiono, and Zheng Zhao

Abstract

The paper analyzes peculiarities of preprocessing of learning data represented in object data bases constituted by multiple relational tables with ontology on top of it. Exactly such learning data structures are peculiar to many novel challenging applications. The paper proposes a new technology supported by a number of novel algorithms intended for ontology-centered transformation of heterogeneous possibly poor structured learning data into homogeneous informative binary feature space based on (1) aggregation of the ontology notion instances and their attribute domains and subsequent probabilistic cause-consequence analysis aimed at extraction more informative features. The proposed technology is fully implemented and validated on several case studies.

Keywords: ontology, object data base, feature aggregation, cause-consequence dependency, non-classical probabilistic space

1. Introduction

The paper proposes automatic feature extraction algorithm in machine learning for classification or recognition. Specificity of the problem statement is that it assumes that learning data (LD) are of large scale and represented in object form, i.e. by multiple tables of relational database with ontology on top of it. Existing techniques for feature extraction and machine learning are mostly oriented to LD represented in the form of a flat table. In case of data stored in object data base, a lot of new problems emerge. Indeed, to extract particular instance (object), it is necessary to use specific query language (Jean et al., 2006). But what is actually challenging here is that various objects can be of various formats and

structures. Every object instance structure is composed of formidable number of concept instances, and each concept can be specified with a lot of heterogeneous attributes, e.g. categorical, Boolean, real valued, and even with a text thus making feature selection and detection of most informative ones a challenging problem.

On the other hand, object data, in its nature, is much more informative in comparison with LD represented in relational data base or in flat table. The main advantage of object-based LD representation is that object data base instances contain *rich context* embedded in it via object structure and object attributes. In fact, each object instance is a piece of knowledge compatible with ontology formalizing meta-knowledge. This is a reason why learning of classification using LD in object form is very perspective and productive although complex research direction.

The paper proposes an original technology for preprocessing of ontology-based LD intended for its transformation into a compact binary-valued flat table representing LD object instances in terms of highly informative features. This technology is demonstrated by a case study, electrical machine diagnosis based of vibro-acoustic data measurements. In the rest of the paper, Section 2 describes briefly the aforementioned case study and its ontology specifying meta-knowledge. Section 3 outlines the proposed technology of ontology-based LD aggregation for feature extraction and filtering. It is worth to note here that the main peculiarity of this filtering algorithm is that the resulting sets of features are *class-specific*. Section 4 outlines the final step of the feature sets and LD transformation to more informative and compressed form via extraction cause-consequence rules. Section 5 concludes the paper and outlines technology perspectives.

2. Cases Study and Domain Ontology

The case study is taken from UCI repository (4, 1990). The task objective is classification of states of electrical pumps using measurements of vibro-acoustic data (VAD) in different measurement points (key points) of pumps in different lines (directions). These data are very multidimensional and have complex structure that is represented by the developed ontology (Fig. 1). Let us describe this ontology while explaining, in parallel, the ontology-based structure of learning data.

For any Electric Pump (EP) *Electric pump* having own *Shaft speed* and state *Machine state*, measurement data *Measuring data* assigned a time stamp is done. This data is presented in the form of VAD *Vibroacoustic measuring*. VADs are measured in several key points *Measure keypoint* along several orthogonal lines (directions). The VADs, in turn, are represented as spectral data *Spectral data* obtained by several filters. As a rule, no more than three filters are used in every key point along every direction. Spectral data are presented by amplitudes *Amplitude* mapped to several values of frequency *Frequency* for current value of time stamp and *Preceding amplitude* in the same key point and along the same line corresponding to the immediately previous value of time stamp. The total number of combinations of used filters and measurement directions is fixed; it is equal to 9. These combinations are introduced as the values of a specific feature "direction-filter" *Direction-Filter*. Example of measurement data instances at a time instant is presented in Fig. 2.

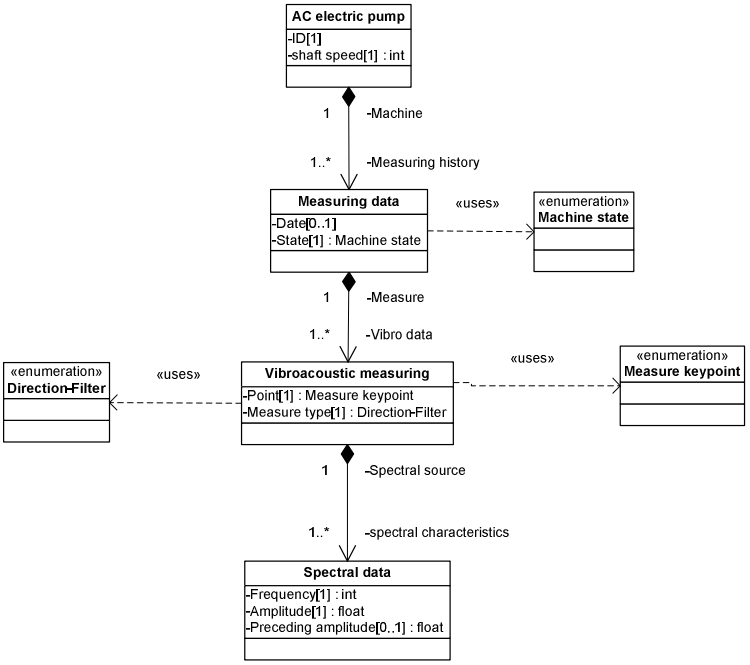


Figure 1: Vibro-acoustic data ontology.

In addition to the above described ontology, so-called ontology of *secondary features* is introduced by the expert. They can be of two categories, *auxiliary features* and *secondary features involved in learning*. Auxiliary features are the ones corresponding to the initial real-valued measurements of spectral data transformed into categorical measurement scale. This transformation was made using overlapping spectral data domain quantization with the total number of intervals equal to 20 with overlapping ratio 10 %. For the secondary features involved in learning, a new feature type is introduced, *Pair-wise of any measurements*. Such feature type contains two positions mapped correspondingly two connected concepts. In general case, components of any pair-wise measurement can be categorical, ordered, or real valued. The secondary feature ontology is given in Fig. 3. In the case study, the following features of the standard or *Pair-wise of any measurements involved in learning* are used:

Secondary features of standard type:

– Nominal amplitude; – Nominal difference of amplitudes.

Secondary features of pair-wise type:

– Frequency–Nominal amplitude
– Frequency–Nominal difference of amplitudes
– Nominal amplitude–Nominal difference of amplitudes
– "Direction-filter" – Nominal difference of amplitudes
– Key point–Frequency
– Shaft speed–Frequency
– Key point–"Direction-filter"
– "Direction-filter" – Nominal amplitude

Let us note that in the case study the components of all secondary features are categorical.

It can be seen that structure of LD, in the case study in question, is rather complex and multidimensional. Due to introduced preliminary expert-based transformation of spectral data it is reduced to a structure of categorical data. Later on, it is used for demonstration of the developed feature extraction procedure.

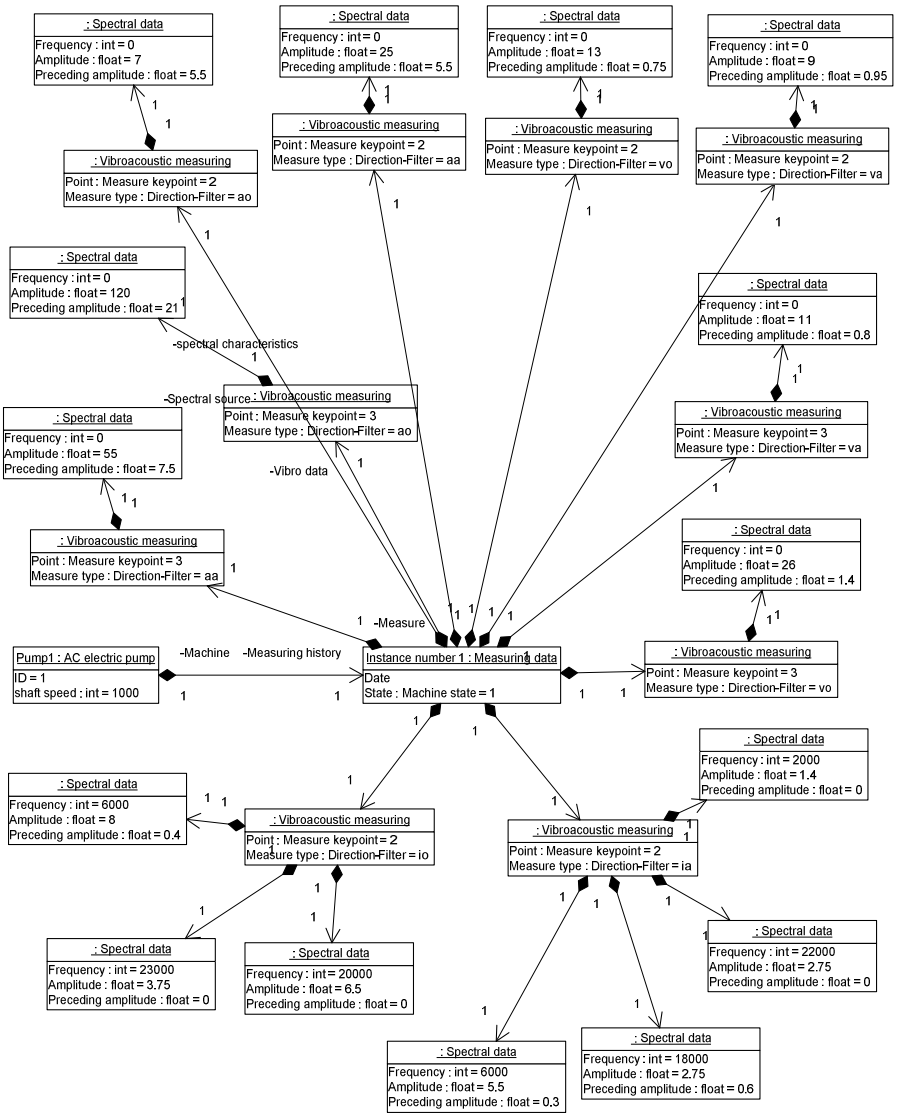


Figure 2: Instance of learning data represented in ontology-based form (in object data base).

3. Technology for Ontology-Based Feature Extraction

3.1 Ontology-Based Learning Data Transformation and Feature Aggregation

The proposed technology is designed for learning of classification with LD stored in object data base. In the case study, such data are structured according to the domain ontology. In general case, LD can also include poorly-structured data in the form of texts on a natural language.

The technology itself illustrated by Fig. 4 is composed of several phases while assuming that ontology can be either given as input information or developed by expert (the last

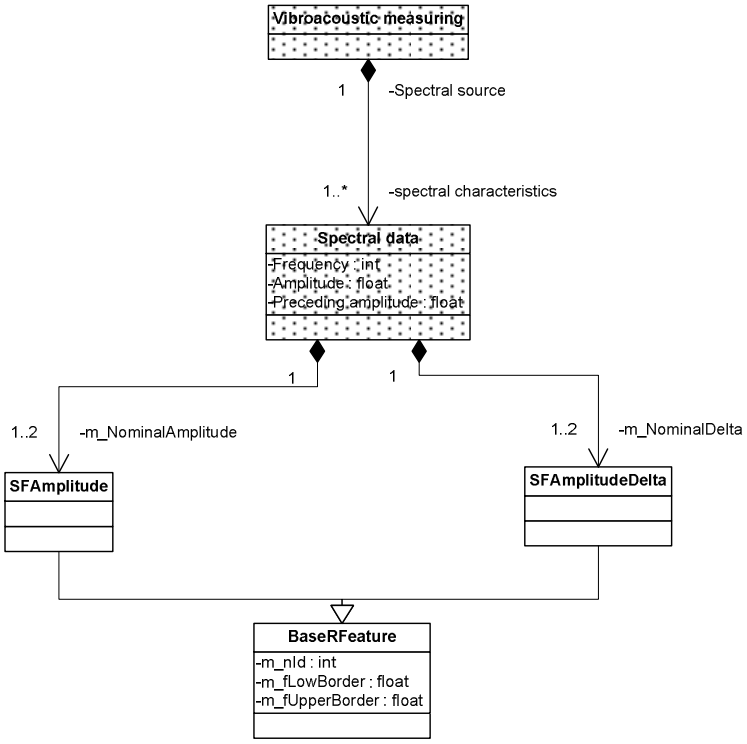


Figure 3: Secondary feature ontology.

case takes place in the case study in question). The *first phase* is expert-based selection of preliminary feature space and transformation of the initial structure of LD to this space. The expert is permitted to select any number of potentially relevant features without any care about types of them or dimensionality, up to thousands. The mandatory requirement here is that the selected features have to be concepts or/and attributes of the ontology. This is important because such features are semantically interpretable and their structure determined by the ontology constitutes particular context of any LD instance. When such preliminary set of features is selected, any instance of LD can be extracted using an object data base query language (Jean et al., 2006). Through such queries all LD instances are transformed into the space of the preliminary selected (potentially redundant) feature space. According to the technology, the resulting LD are represented as *"star"-structured* set of tables, in which columns of fact tables corresponds to elements of the designated preliminary feature set with one row in kernel table per every LD instance assigned a class label. This representation is context-dependent where different LD instances can be of different formats since some features introduced by expert can be irrelevant to particular instances. Therefore any table of star structure can contain "missing-like" values to be interpreted as *"irrelevant"* to the corresponding object instance.

The *second phase* is aggregation and filtering of the features selected at the first phase, as well as representation of filtered set of aggregated features in unary predicate form. The final procedure of the second phase is transformation of LD obtained at phase 1 to new *class-centered feature space*. Let us briefly explain the mathematical idea of feature

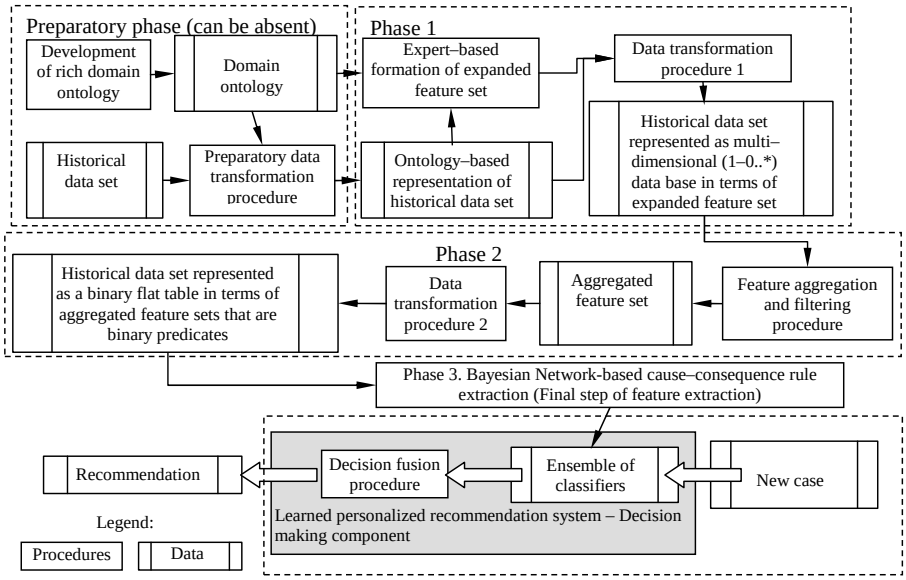


Figure 4: Ontology-based classification system technology.

aggregation procedure while using following denotations: $\Omega = \{\omega_1, \ldots, \omega_m\}$ stands for the set of classes labels that can be assigned to an LD instance, e.g. $\Omega = \{1, \ldots, 6\}$, in the case study; $\mathbf{X} = \{X_1, \ldots, X_n\}$ — the set of feature *identifiers* (ID), where X_i stands for particular feature ID; x_s^i — particular value of the feature with ID X_i, and $\aleph_i$ — domain of the feature X_i, i.e. $x_s^i \in \aleph_i$. Let us note that cardinality of any feature X_i domain may be huge (if either categorical, or numeric, or real valued). For example, categorical feature *"Key male role in a movie"* in the NetFliX task (5) can possesses thousands of values corresponding to particular actors' names. Let also symbol $\sum$ stands for the set of LD instances in the target *filtered feature space*.

Feature aggregation and filtering is realized by single procedure. For a value x_s^i of feature X_i, $x_s^i \in \aleph_i$ and a class ω_k, an aggregate $\aleph_i(\omega_k) \in \aleph_i$ is defined as follows:

$$x_s^i \in \aleph_i(\omega_k) \text{ if and only if for } \forall \omega_\nu \in \Omega,\ \nu \neq k:\ p(\omega_k/x_s^i) > p(\omega_\nu/x_s^i) + \Delta, \qquad (1)$$

where Δ is a positive real value defining a dominance threshold. The inequality (1) states that conditional probability of the class ω_k, $p(\omega_k/x_s^i)$, if the feature X_i is instantiated by the value x_s^i is larger than the same conditional probability for any other class. Thus, to compute an aggregate $\aleph_i(\omega_k)$, it is necessary to check the inequality (1) for $\forall x_s^i \in \aleph_i$ and $\forall \omega_\nu \in \Omega$ for all $\nu \neq k$. Each such aggregate can be computed using sample $\sum$.

Finally, at the second phase, let us introduce unary predicates $B_i(\omega_k)$ that are instantiated by the truth value *"true"* if and only if $x_s^i \in \aleph_i(\omega_k)$, and *"false"*, otherwise. The truth domains of these predicates are determined uniquely by aggregates with the same subscripts and argument ω_k values. Thus, the results of the second phase are the aggregates $\aleph_i(\omega_k)$ and corresponding unary predicates $B_i(\omega_k)$, $i \in I(\omega_k)$, where $I(\omega_k)$ is the subset of indexes of features X_i successfully passed the test (1) for fixed ω_k.

Using inequality (1) and definition of the predicates $B_i(\omega_k)$, the LD sample $\sum$ is transformed to the set of samples $\sum(\omega_1), \ldots, \sum(\omega_m)$, representing LD in the space of binary features that are predicates $B_i(\omega_k)$.

The authors' experience based on prototyping of several applications where the developed technology was used showed that, as a rule, the procedure (1) filters many features of the set $\{X_1, \ldots, X_n\}$ that are not satisfied with (1) for any $\omega_k \in \Omega$. Let us also note that the value Δ of the dominance threshold can be used as a means to restrict the total number of the finally extracted features (either aggregates $\aleph_i(\omega_k)$, or unary predicates $B_i(\omega_k)$) to a predefined limit.

Thus, in result of the phases 1 and 2 the source high dimensional heterogeneous LD of a complex structure is transformed to a homogeneous binary feature space of desirable dimension.

3.2 Cause-Consequence Rule Extraction

Phase 3 starts when aggregates $\aleph_i(\omega_k)$, unary predicates $B_i(\omega_k)$, $i \in I(\omega_k)$, $\omega_k \in \Omega$ and LD samples $\sum(\omega_1), \ldots, \sum(\omega_m)$ are formed. In Fig. 4 this phase is denoted as phase 3. Its objective is to find cause-consequence dependencies (rules) between conjunction of predicates $B_i(\omega_j)$, $i \in I(\omega_j)$, $\Omega = \{\omega_1, \ldots, \omega_m\}$ and $\omega_j \in \Omega$. For this purpose, a probabilistic approach is used. Let us describe it for particular $\omega_k \in \Omega$.

For $\omega_k \in \Omega$ probabilistic space is introduced as follows. The set of aggregates $\aleph_i(\omega_j)$ is considered as a family set $\{\aleph_i(\omega_j)\}|_{i \in I(\omega_k), j=1,\ldots,m}$, where each set $\aleph_i(\omega_j)$ is mapped a probabilistic measure

$$p(\aleph_i(\omega_j)) = |\aleph_i(\omega_j)|/|\aleph_i|, \tag{2}$$

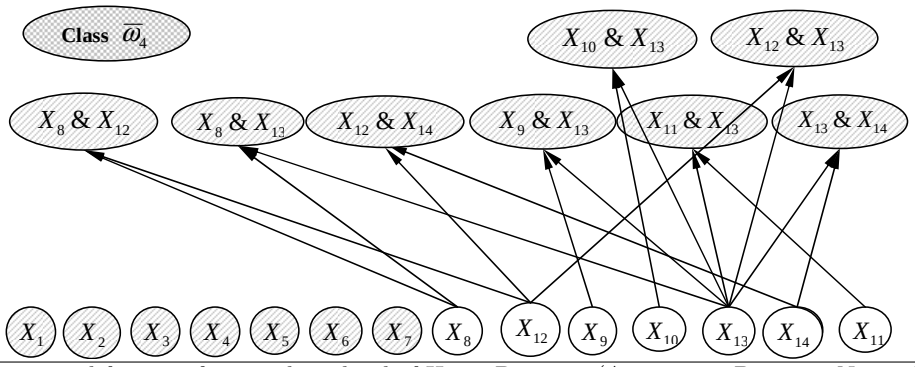
where $|\ |$ denotes cardinality of the corresponding set. It is clear that

$$p(B_i(\omega_j)) = p(\aleph_i(\omega_j)) \tag{2'}$$

Since the aggregates $\aleph_i(\omega_j)$ can overlap with $\aleph_i(\omega_r)$, $j \neq r$ these aggregates and corresponding *random events* can be *dependent*. Each set $\aleph_i(\omega_j)$ of the family set $\{\aleph_i(\omega_j)\}|_{i \in I(\omega_k), j=1,\ldots m}$, can also be correlated with any $\omega_k \in \Omega$, which are also considered in the model as random events with predefined a *priory probabilities*. Therefore the sets of family $\{\{\omega_k\}|_{k=1}^m, \{\aleph_i(\omega_j)\}|_{i \in I(\omega_k), j=1,\ldots m}\}$, cannot be used as elementary random events and thus probabilistic space cannot be defined here in the classical manner. In this work, "non-classical" definition of the probabilistic space and corresponding non-classical probability space axiomatics are used (Halpern, 2003).

While omitting some algebraic details, this probabilistic space projected to the subspace taking into account only ω_k can be modeled as an upper $\gamma_k^{\vee} =< \{\omega_k, \{\aleph_i(\omega_j)\}|_{i \in I(\omega_k), j=1,\ldots m}\}$, $\geq>$ or lower $\gamma_k^{\wedge} =< \{\omega_k, \{\aleph_i(\omega_j)\}|_{i \in I(\omega_k), j=1,\ldots m}\}, \leq>$ semi-lattice, where order relation is defined in usual theoretic-set sense. In this semi-lattices, any node is mapped a probability of the corresponding random event. Further on, the lower semi-lattice is used. In this semi-lattice, ω_k is the class label node called below *"target node"*. The model described below is identical for any $\omega_k \in \Omega$.

Definition. Hasse diagram of the lower semi-lattice $\gamma_k^{\wedge} =< \{\omega_k, \{\aleph_i(\omega_j)\}|_{i \in I(\omega_k), j=1,\ldots m}\}$, $\leq>$ is below called Associative Bayesian Network (ABN).



Aggregated features forming base level of Hasse Diagram (Associative Bayesian Network)		
X_{14}: NAmplitude-NDelta_0_2	X_6: VibroFrequency-NAmplitude_0_4	X_{11}: NAmplitude_0_3
X_8: VibroFrequency-NDelta_0_5	X_7: VibroFrequency-NDelta_0_4	X_{10} : MType-VibroFrequency_0_2
X_3 : MType-VibroFrequency_0_4	X_2: PShaftSpeed-VibroFrequency_0_4	X_1: MType-NDelta_0_4
X_4: Point-VibroFrequency_0_4	X_{12}: VibroFrequency-NAmplitude_0_2	X_9: NDelta_0_3
X_5: MType-NAmplitude_0_4	X_{13}: VibroFrequency-NDelta_0_2	

Figure 5: Example of Associative Bayesian Network for case study, representing cause — consequence rules for class $\overline{\omega}_4$.

Let us note that this notion was introduced in the paper (Gorodetski, 1992). Fig. 5 gives an example of a fragment of ABN built for the case study described in Section 2. Semantics of the aggregates is also described in that section.

Let us consider the set $\{\omega_k, \{B_i(\omega_j)\}|_{i\in I(\omega_k),j=1,...m}\}$ as the basic set (nodes) of the lower semi-lattice $\gamma_k^{\wedge}$ that is isomorphic with the semi-lattice $\gamma_k^{\wedge} =< \{\omega_k, \{\aleph_i(\omega_j)\}|_{i\in I(\omega_k),j=1,...m}\}, \leq >$. General idea of the developed algorithm of cause-consequence (CC)— rule extraction consists in iterative construction of ABN which nodes represent premises of the cause-consequence rules (CC-rules) in the form *<conjunction of a subset of the basic predicates of ABN with negation or without it>* $\Rightarrow \omega_k$ only. This algorithm is iterative and the number of particular iterations coincides with the length of conjunctive premises generated at corresponding iteration. Below very short and slightly simplified outline of CC-rule extraction algorithm is done (due to limit of the paper space). Below the denotation $\widehat{B_i}(\omega_j)$ is used predicate identifier (literal) that can take two values: $B_i(\omega_j)$ if it is considered without negation and $\overline{B_i}(\omega_j)$ with negation.

1. *Generation of the rule set containing 1-literal premises.* Let $\{\omega_k, \widehat{B_i}(\omega_j)\}|_{i\in I(\omega_k),j=1,...m}$ be all the pairs composed of a literal $\widehat{B_i}(\omega_j)|_{i\in I(\omega_k),j=1,...m}$ and the target node ω_k. The first to be done is to assess joint probabilities $p(\widehat{B_i}(\omega_j)\omega_k)$ for every assignment of the literal $\widehat{B_i}(\omega_j)$. Three filters applied to $p(\widehat{B_i}(\omega_j)\omega_k)$ described below are sequentially used to filter the above pairs that can be the sources of rules in the form *"If* $\widehat{B_i}(\omega_j)$ *then* ω_k*"* assigned confidence measure $p(\omega_k/\widehat{B_i}(\omega_j))$ where $\widehat{B_i}(\omega_j) \in \{B_j(\omega_j), \overline{B_i}(\omega_j)\}$ (positive and negative literals respectively).

Filter 1 (filters the rules containing independent premises and consequents)

$$I(B_i(\omega_j), \omega_k) = |p(B_i(\omega_j)\omega_k) - p(B_i(\omega_j))p(\omega_k)|/[p(B_i(\omega_j))p(\omega_k)] \geq \delta_{\min} > 0 - a \qquad (3)$$

selection threshold. Otherwise, the corresponding 1-literal rule is non-interesting.

Filter 2 (filters the rules that are dependent but do not correspond to the CC-dependencies)

$$R(\widehat{B_i}(\omega_j),\omega_k) = |p(\omega_k/\widehat{B_i}(\omega_j)) - p(\omega_k/\widehat{B_i}(\omega_j))|/\{p(\widehat{B_i}(\omega_j)))[1-p(\widehat{B_i}(\omega_j))]\} =$$

$$= |p(\widehat{B_i}(\omega_j)\omega_k) - p((\widehat{B_i}(\omega_j)))p(\omega_k)|/\{p(\widehat{B_i}(\omega_j))[1-p(\widehat{B_i}(\omega_j))] \geq \delta_{\min}, \delta_{\min} > 0 - a \quad (4)$$

selection threshold value, $\widehat{B_i}(\omega_j) \in \{B_i(\omega_j), \widehat{B_i}(\omega_j)\}$; Otherwise, the corresponding 1-literal rule is non-interesting.

Notice: In fact, this filter is more complex. The filtration has to be done not only for any possible assignments of random event $\widehat{B_i}(\omega_j) \in \{B_i(\omega_j), \overline{B_i}(\omega_j)\}$, but also for two assignments of random event $\widehat{\omega_k} \in \{\omega_k, \overline{\omega_k}\}$ in order not to lose the rules in the form $\widehat{B_i}(\omega_j) \Rightarrow \overline{\omega_k}$. If , at least, for one of variant of assignment of above mentioned random events the filtration is successful then corresponding 1-literal rule remains to be a candidate, otherwise it is deleted from the candidate set. Here and at the subsequent steps of CC-rule extraction such additional checks are assumed on default and are not described due to limitation of the paper space.

Let us note that measure $R(\widehat{B_i}(\omega_j),\omega_k)$ is well known in probability theory and mathematical statistics as *regression coefficient of the random events* $\widehat{B_i}(\omega_j)$ and ω_k.

Filter 3 (filters CC-rules with low confidence)

$$p(\widehat{\omega_k}/\widehat{B_i}(\omega_j)) = p(\widehat{B_i}(\omega_j)\widehat{\omega_k})/p(\widehat{B_i}(\omega_j)) \geq \gamma_{\min} \quad (5)$$

at least, for one of assignments of the random events $\widehat{B_i}(\omega_j)$ and $\widehat{\omega_k}, \gamma_{\min} > 0 - a$ selection threshold value. Otherwise, the corresponding 1-literal rule is non-interesting

Let us denote the chosen set of 1-literal premises as C_1. It is a set of literals $\widehat{B_i}(\omega_j)$, $i \in I_1(\omega_j)$ that remain to be the candidates of potential 1-literal premises of the rules $\widehat{B_i}(\omega_j) \Rightarrow \widehat{\omega_k}$ or premises of more length. Other literals $\widehat{B_i}(\omega_j) i \notin I_1(\omega_j)$, are not anymore considered in the subsequent algorithm.

2. *Generation of the rule set containing 2-literal premises.* In general, this step is about the same as the previous step with the two differences. Due to limit of the paper space, let us not describe this and subsequent steps but formulate only the differences.

First difference is that, at this step, all the conjunctive pairs $\widehat{B_i}(\omega_j) \wedge \widehat{B_j}(\omega_r)$, $\widehat{B_i}(\omega_j)$, $\widehat{B_j}(\omega_r) \in C_1$ are considered as the 2-literals CC-rule premises candidates. They are subjected to the analogous three step filtration used for 1-literal rules, and then, like C_1, the set C_2 of 2-literals premises containing the chosen conjunctive pairs $\widehat{B_i}(\omega_j) \wedge \widehat{B_j}(\omega_r) \in C_2$, $i, j \in I_2(\omega_k)$ is formed (ω_k is target node).

Second difference is that additionally, at this (and, in analogy, at the subsequent steps too), the set of non-chosen predicate literals $\widehat{B_i}(\omega_j)$ are united in the set $\mathcal{A}_1(\omega_k)$ that is the set of 1-literal premises of the rules $\mathcal{R}_1(\omega_k)$ in the form $\widehat{B_i}(\omega_j) \Rightarrow \omega_k$, $i \in I_1(\omega_k)$, containing ω_k in the consequences.

The process stops when either the set C_k of k-literals candidates became empty, or a predefined number of rules is found. The latter often is a good choice in order to prevent an over-fitting. Control attribute Δ in the equation (1) plays the same role. The resulting set $\mathcal{A}(\omega_k) = \bigcup_{r=1}^{N} \mathcal{A}_r(\omega_k)$, is the target set of features forming new feature space.

4. Some Experimental Results

An extended experiment was performed for the case study described in Section 2. Let us first note that in UCI repository (4, 1990) only results obtained by the benchmark authors are given. In fact, this task is too complex for existing approaches due to very complex data structure. Unfortunately the benchmark contains very limited number of instances (objects instances). They were divided into training and testing sets and the latter were not involved in learning procedure. The results of testing of the produced classifier on training data are presented in Tab. 1, whereas the results of its testing on the data that was not used in training are done in Tab. 2. Let us comment shortly these results. It is important to note that training data set has much less training instances as compared with testing one. One of our ideas of such decision was to check performance of the developed feature selection technology on relatively small training sample. It can be seen from the Tab. 1 that classification quality with regard to training sample is rather good. What concerns testing sample, it is important to note that the resulting algorithm has practically no misclassification, but in a large number of cases it refused to decide in favor of particular class. But classification algorithm was not carefully designed due to the fact that the paper objective is other than design good classification algorithm. More important, for this paper, is that the features designed according to the proposed technology found out informative and even for not carefully designed classifier provides the decision quality that is not worth in comparison of the results provided in UCI repository.

Table 1. Contingency matrix for testing of classifier on training data

	1	2	3	4	5	6	Refusal
1	15						1
2		16					
3			7				1
4				10			
5					10		
6						15	1

Table 2. Contingency matrix for testing of classifier on new data

	1	2	3	4	5	6	Refusal
1	17						46
2		16					53
3			7				7
4	1			10			7
5					10		7
6						15	13

5. Conclusion

The authors' practical experience proved that the proposed feature space synthesis approach works well in very "heavy" high dimensional learning tasks using heterogeneous relational data with ontology on top of it. One of the important advantages of the developed approach is that the resulting feature space is homogeneous (binary) and most of the existing classification mechanisms can be used at decision making step. The proposed feature extraction approach was fully implemented and validated using several applications. It was also used in design and implementation of an ontology-based profiling and recommending system. In particular, intelligent e-mail assistant for incoming e-mail sorting was prototyped.

References

Netflix. http://www.netflix.com. different pruning measures can be used but this aspect is out of the paper scope.

Mechanical analysis data set, machine learning uci repository, 1990. URL `http://archive.ics.uci.edu/ml/datasets/Mechanical+Analysis`.

V. Gorodetski. Adaptation problems in expert systems. *International Journal of Adaptive Control and Signal Processing*, 6:201–209, 1992.

J. Y. Halpern. *Reasoning about uncertainty*. MIT Press: Cambridge, 2003.

S. Jean, Y. Ait-Ameur, and G. Pierra. Querying ontology based database using ontoql (an ontology query language). In *LNCS*, volume 4275, pages 704–721. Springer, 2006.