Evaluation of GUHA Mining with Background Knowledge

Martin Ralbovský

Department of Information and Knowledge Engineering,
University of Economics, Prague, W. Churchill Sq. 4, 130 67 Praha 3, Czech Republic
martin.ralbovsky@gmail.com

Abstract. Background knowledge

Keywords: Background knowledge, GUHA Method, STULONG dataset

1 Introduction

Process of knowledge discovery in databases (KDD) can be affected by using domain knowledge. In [18] authors identified four KDD stages: data understanding, task design, result interpretation and result dissemination over the semantic web, where proper domain knowledge (ontologies) can be helpful. In this work, we are interested in evaluation of KDD techniques with respect to the used domain knowledge and examined data. The evaluation should help to improve the task design and result interpretation KDD stages.

We are using the STULONG[1] database as the examined data. The STULONG database is an extensive epidemiological study of atherosclerosis primary prevention and was examined also in [18]. Besides the data, STULONG contains some domain knowledge examples created by medical experts. The knowledge (here named *background knowledge*) consists of verbal rules expressing relationships between two entities in the domain.

Because of the fact, that most of the data mining analysis with STULONG were done with tools implementing GUHA method, we chose this method to be evaluated by the background knowledge. By evaluation we mean constructing various data mining tasks that should approve or disapprove the background knowledge in the STULONG data and drawing conclusions from the results of the tasks. We invented a formalization of verbal background knowledge rules and implemented automatic tools to verify them against the outputs of GUHA mining tasks. To our best knowledge, this work is the first work to evaluate GUHA mining on bases of comprehensive background knowledge verification.

The work is structured as follows: section 2 describes the GUHA method, GUHA procedures used in this work and also recent tools implementing the GUHA method. Section 3 explains background knowledge used, new formalization of the background knowledge and example of the formalization. 4 shows

[1] http://euromise.vse.cz/stulong

conducted experiments evaluates the GUHA method on basis of the experiments. Section 5 puts the work into context of other works dealing with *background knowledge* and section 6 concludes the work and gives ideas about future research.

2 The GUHA Method

GUHA method is one of the first methods of exploratory data analysis, developed in the mid-sixties in Prague. It is a general mainframe for retrieving interesting knowledge from data. The method has firm theoretical foundations based on observational calculi and statistics [6], [7]. For purpose of this work let us explain only the basic principles of the method, as shown in Figure 1.

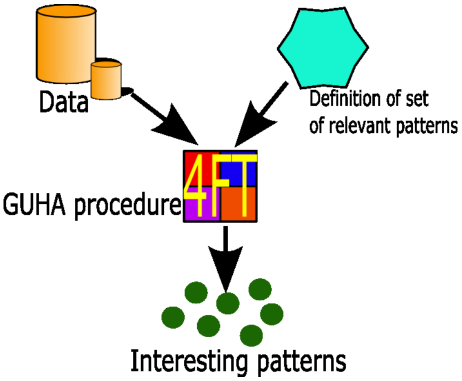


Fig. 1. The GUHA method

GUHA method is realized by GUHA procedures such as 4FT procedure to be described, located in the middle of the figure. Inputs of the procedure are data and a simple definition of a possibly large set of relevant patterns, which will be discussed in detail in the following section **??**. The procedure automatically generates all the relevant patterns and verifies them against the provided data. Patterns that are true are output of the procedure. In this work, we use procedure 4FT (described in section 2.1) and procedure KL (described in section 2.2).

Definition 2 *Each basic Boolean attribute is a* **Boolean attribute**. *If α and β are Boolean attributes, $\alpha \wedge \beta$, $\alpha \vee \beta$ and $\neg\alpha$ are* **Boolean attributes**.

2.1 Procedure 4FT

Classical *apriori* [1] association mining searches rules in form $X \longrightarrow Y$, where X and Y are sets of items. Procedure 4FT searches (in the simplified form) for generalized association rules in form $\varphi \approx \psi$, where φ and ψ are *Boolean attributes* and $\approx$ is a *4ft-quantifier*[2]. Relation $\varphi \approx \psi$ is evaluated on the basis of *4ft table*, as shown in Table 1.

The term *Boolean attribute* is created from attributes. We use the term attribute in the sense of *categorial attribute*, i.e. attribute with finite number of values. Let A be an attribute, A $= \{a_1, a_2 ... a_n\}$ and $\alpha \subset A$, $\alpha \neq \emptyset$. Then $A(\alpha)$ is a *basic Boolean attribute*.

M	ψ	$\neg\psi$
φ	a	b
$\neg\varphi$	c	d

Table 1: 4ft table

Table 1. 4FT contingency table

A *4ft table* is a quadruple of natural numbers $\langle a,\ b,\ c,\ d \rangle$ so that:

- a: number of objects (rows of M) satisfying φ and ψ
- b: number of objects (rows of M) satisfying φ and not satisfying ψ
- c: number of objects (rows of M) not satisfying φ but satisfying ψ
- d: number of objects (rows of M) satisfying neither φ nor ψ

4ft-quantifier expresses kind of dependency between φ and ψ. The quantifier is defined as a condition over the *4ft table*. In this work, we use the two most common *4ft-quantifiers*: *founded implication* and *above average dependence*.

The *founded implication* is the basic quantifier for the 4FT procedure. It is defined by the following condition:

$$a \geq Base \wedge \frac{a}{a+b} \geq p$$

where $Base$ and p are threshold parameters of the procedure. The $Base$ parameter represents absolute number of objects that satisfies φ. In our work we will use relative $Base$ representation, $\frac{a}{a+b+c+d}$. The $Base$ parameter corresponds to the *support* and p to the *confidence* parameters of classical association mining.

The *above average dependence* is defined by the following condition:

$$\frac{a}{a+b} \geq (p) \frac{a+c}{a+b+c+d} \wedge a \geq Base$$

where p and $Base$ are user-defined parameters[3]. Again, we will use the relative $Base$ representation $\frac{a}{a+b+c+d}$. So, the quantifier can be verbally interpreted as

[2] The more complex form includes another *Boolean attribute* as a condition In our work we do not mine for conditional rules, therefore we omit the more complex definition.

[3] The p parameter is originally defined in [14] as $\frac{\mathsf{a}}{\mathsf{a+b}} \geq (1+p)\frac{\mathsf{a+c}}{\mathsf{a+b+c+d}}$. We alter this definition in order to avoid negative p results in the experiments.

Among object satisfying φ, there are at least p per cent more objects satisfying ψ then among all observed objects and there are at least Base per cent of observed objects satisfying φ and ψ.

2.2 Procedure KL

Procedure KL [15] searches (in the simplified form) for rules in form $R \sim C$, where R and C are *categorial attributes*[4]. The symbol $\sim$ is called *KL-quantifier.* The rule $R \sim C$ means, that *categorial attributes* R and C are in relation described by $\sim$. In this work, we are using the *Kendall's quantifier.*

Kendall's quantifier is based on *Kendall's coeficient* τ_b[17]. It is defined as

$$\tau_b = \frac{2(P-Q)}{\sqrt{(n^2 - \sum_k n_{k,*}^2)(n^2 - \sum_l n_{*,l}^2)}}$$

where

$$P = \sum_k \sum_l n_{k,l} \sum_{i>k} \sum_{j>l} n_{i,j},\, Q = \sum_k \sum_l n_{k,l} \sum_{i>k} \sum_{j<l} n_{i,j}$$

τ_b ranges from $\langle -1, 1\rangle$, where values $\tau_b > 0$ indicate positive ordinal dependence[5], values $\tau_b < 0$ negative ordinal dependence, $\tau_b = 0$ ordinal independence and $|\tau_b| = 1$ functional dependence of C on R. In this work, we are using the *Kendall's quantifier* to construct *abstract quantifiers* discussed in section 3.1.

2.3 GUHA Tools

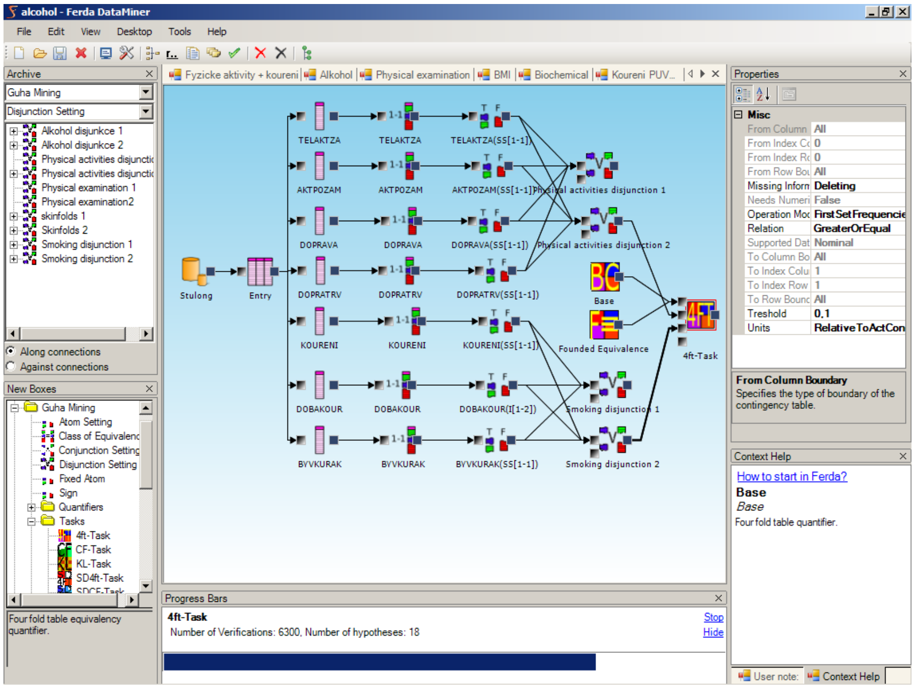
Apart from the tools presented in this section, several systems implementing GUHA procedures were developed in the past. In recent years, the *LISp-Miner* system has been the most significant GUHA tool. This system has been under development since 1996 at the University of Economics, Prague. The system includes six GUHA procedures including procedure KL and lighter version of 4FT procedure [13] in addition to other data preparation and result interpretation modules.

In 2004, the Ferda project started as an initiative to build a new visual data mining successor of the *LISp-Miner* system. Creators (at the Faculty of Mathematics and Physics, Charles University, Prague) succeeded in developing an user friendly visual system with advanced features such as high level modularity, support for distributed computing or reusability of the task setting [8]. At present there are several research activities taking advantage of the system.

Figure 2 shows the Ferda working environment. For purposes of this work, there were modules implemented in the Ferda system as well.

[4] Again, there can be *Boolean attribute* added as a condition to the rule.

[5] High values of C often coincide with high values of R, low values of C often coincide with low values of R.



Four fold table equivalency quantifier.

4ft-Task

Number of Verifications: 6300, Number of hypotheses: 18

From Column	All
From Index Cc	0
From Index Rc	0
From Row Bou	All
Missing Inforn	Deleting
Needs Numeri	False
Operation Moc	FirstSetFrequencie
Relation	GreaterOrEqual
Supported Dat	Nominal
To Column Bo	All
To Index Colu	1
To Index Row	1
To Row Bounc	All
Treshold	0,1
Units	RelativeToActCon

From Column Boundary
Specifies the type of boundary of the contingency table.

How to start in Ferda?

Base
Base
Four fold table quantifier.

Fig. 2. Ferda environment

3 Background Knowledge

3.1 Considered Background Knowledge Types

Background knowledge (also *field knowledge* or *prior knowledge*) is knowledge that comes from the user or a community of users and integrates knowledge, which the community can agree upon and consider it common. Various fields of KDD define background knowledge differently, there is no central theory for the term. In the context of GUHA mining, we think of background knowledge as a part of domain knowledge, knowledge that is specific to particular domains (medicine, chemistry, etc.). We define background knowledge as a set of various verbal rules that are accepted in a specific domain as a common knowledge[6]. The rule can describe functional dependence between quantities, relationship between entities or say something about their behavior. Below are presented example rules taken from STULONG[7]:

– If education increases, wine consumption increases as well.

[6] Note the difference between our vague definition and precise definitions e.g. in ILP

[7] The study (STULONG) was realized at the 2nd Department of Medicine, 1st Faculty of Medicine of Charles University and Charles University Hospital, U nemocnice 2, Prague 2 (head. Prof. M. Aschermann, MD, SDr, FESC), under the supervision of Prof. F. Boudk, MD, ScD, with collaboration of M. Tomekov, MD, PhD and Ass.

– Patients with greater responsibility in work tend to drive to work by car.

3.2 Background Knowledge Formalization

In order to automatically verify background knowledge against the data, a new formalization needed to be thought out. Background knowledge contains heterogeneous verbal formulations of dependences and relationships in the domain. The relevance and validity of the formulations varies: the relationships in physics are formed exactly by mathematical equations, but for example in sociology they mean only expected behavior or opinion of a group of people. Our aim is to find formalization usable for both domains.

We present a new *Formalization with attributes, validation literals and abstract quantifiers* first used in [12] The main idea behind the formalization is to make it as close to GUHA terms as possible while still enabling large expressive possibilities of the verbal rule. Because of shorter format of the article, we present only an overview and an example of the new formalization with a little reasoning. The topic is fully covered in [12], section 3.2.2.

Attribute is the basic term for the new formalization. *Attribute* is defined as a result of domain categorization and is used to create *categorial attributes*, inputs of the *KL* procedure.

Validation literal is a special type of *literal* used to express background knowledge. *Literal* is a *basic Boolean attribute* or its negation. We define the *literal length* as the size of the categories' subset. *Validation literal* is a literal, which has *literal length* equal to 1.

Abstract quantifier is a generalization of a quantifier or quantifiers of a procedure (4FT or KL). The idea behind *abstract quantifiers* is to create a "black-box" quantifier: user does not need to fill any numeral parameters of the quantifier. The quantifier is then more suitable for transferring verbal *background knowledge* rules into formalized form.

3.3 Formalization Example

With all the terms explained, let us see how the formalization is applied to a specific verbal rule **If education increases, wine consumption increases as well.** as presented in Section 3.1. The rule defines relationship between two measurable quantities of a patient. These quantities are stored in the database in the form of columns of a table, so *attributes* can be created. We name the attributes for **education** and **wine consumption** *education* and *wine* respectively.

Prof. J. Bultas, MD, PhD. The data were transferred to the electronic form by the European Centre of Medical Informatics, Statistics and Epidemiology of Charles University and Academy of Sciences (head. Prof. RNDr. J. Zvrov, DrSc). The data resource is on the web pages `http://euromise.vse.cz/challenge2004`. At present time the data analysis is supported by the grant of the Ministry of Education CR Nr LN 00B 107.

For this paragraph we will consider only the *KL* procedure. The procedure searches (in the simplified form) for rules in form $R \sim C$, where R and C are *categorial attributes*, which derive from *attributes*. When **education** and **wine consumption** out of the rule are to be formalized with R and C of the hypothesis, then the part **If ... increases, ... increases as well** could be formalized with a proper *abstract quantifier*. We call this quantifier *increasing dependence* and is implemented as a special setting of the *Kendall quantifier* (to be described later). With all the knowledge stated above, the rule **If education increases, wine consumption increases as well** can be formally written as *education* $\uparrow$ *wine*, where $\uparrow$ states for *increasing dependence abstract quantifier*.

We can also define the formalization for the *4FT* procedure. The hypotheses of this procedure consist of *Boolean attributes*, therefore it is better to use *validation literals*. If we presume correct categorization, out of *attributes education* and *wine* the *validation literals education(HIGH)* and *wine(HIGH)* can be created[8]. Similarly to *KL* formalization we can use *abstract quantifier* to note the dependence. Then the rule **If education increases, wine consumption increases as well** can be formalized as *education(HIGH)* $\Rightarrow$ *wine(HIGH)* with a proper *abstract quantifier* $\Rightarrow$

Formalization with the *4FT* procedure does not consider the whole *attributes* but only some of its categories, thus it is weaker. However there may be situations when it is feasible to use the *4FT* procedure. If the examined attribute is not ordinal, the KL procedure cannot be used. Also there may be ordinal attributes with such a small number of categories, that is preferred to use 4FT procedure[9].

In the beginning of this section, a requirement was given on the formalization to be able to represent various kinds of relationships between the entities of the domain. The *formalization with attributes, validation literals and abstract quantifiers* fulfills this requirement, because the formalization does not pose any restrictions on the relationships - the relationship is expressed by the *abstract quantifier*.

4 Experiments

The main reason for constructing a formalization was to experimentally find out, if background knowledge gained from domain experts is apparent in the data by GUHA means. This part of the paper gives information about experiments: section 4.1 describes experiments' setup, sections 4.2 and 4.3 show two conducted experiments and section 4.4 discusses the results of the experiments.

4.1 Setup

Modules of the Ferda system were created for purposes of this work and of work [12]. The modules enable the *formalization with attributes, validation literals and*

[8] *Validation literal* allows sign setting. Here both signs are *positive*.

[9] In our experiments that was often the case

abstract quantifiers setting. They also automatically find rules from the output of 4FT and KL procedures that match the formalized background knowledge. The details of the implementation, with proper explanation of the modules and description of algorithms can be found in [12].

Special attention was paid to selection of *abstract quantifiers*. For the KL procedure, we chose variations of *Kendall quantifier* named *increasing* and *decreasing dependence* for observing positive and negative ordinal dependence. Out of many *4ft-quantifiers*, we chose the two most used quantifiers introduced in section 2.1, the *founded implication* and *above average dependence*. We presumed that if they are most used, they should be somehow "good".

We chose 8 sample background knowledge rules concerning education and responsibility in work. These rules were selected as a sample of the rules that can be mined upon (without changing the database schema). Rules are listed in Table 2. We used the same common categorization of STULONG attribute both for the task settings and for the formalization settings.

Number	**Rule - left side**	**right side**
1	If education increases	physical activity after work increases as well
2	If education increases	responsibility in work increases as well
3	If education increases	wine consumption increases as well
4	If education increases	smoking decreases
5	If education increases	physical activity in work decreases
6	If education increases	beer consumption decreases
7	Patients with greater responsibility in work	tend to drive to work by car
8	Patients with smaller responsibility in work	tend to use public transport to get to work

Table 2. Verified rules

4.2 Default Quantifiers' Settings

There are threshold values of parameters defined for each quantifier, which tell us when quantifier's output is significant. We call them *default quantifiers' settings*. These values were set up by an agreement among data mining experts [10]. The aim of the first conducted experiment was to verify, if there are in the data any rules found with aid of formalization and *abstract quantifiers* defined in previous section backing the background knowledge.

We chose 0.7 and -0.7 value of the *Kendall's coeficient* for the *increasing* and *decreasing dependency abstract quantifiers* respectively. For the *founded implication* quantifier, the default values are 0.95 for the p parameter and 0.05 for

[10] However, the values are not fixed and can be subject of further discussion

the (relative) *Base* parameter. For the *above average dependence* quantifier, the default values are 1.2 for th p parameter and again 0.05 for the *Base* parameter.

Rule number	**ID**	**DD**	**FI**	**AA**
1	YES	x	NO	NO
2	YES	x	NO	NO
3	NO	x	YES	NO
4	x	NO	NO	NO
5	x	NO	NO	YES
6	x	NO	NO	NO
7	x	x	NO	NO
8	x	x	NO	NO

Table 3. Verification of quantifier's settings

Table 3 shows the results of the first experiment. The **ID**, **DD**, **FI** and **AA** stands for *increasing dependence*, *decreasing dependence*, *founded implication* and *above average* quantifiers. **YES** means that the rule was found with the given quantifier, **NO** means that the rule was not found and **x** means that the rule was not meaningful for the given quantifier.

Before we draw any conclusions from the experiment, let us first state some presumptions about the data source. The data table *Entry*, which was mined upon, contains records about the entry examination of 1417 patients. Because of this number, we consider the data to be statistically significant. We also presume no errors in the data and proper categorization (described in [12]). Finally, when we want to question settings of individual quantifiers, we presume that the background knowledge rules are "somehow stored" in the data. For example that the number of patients approving the background knowledge rule is greater then the number of patients disapproving the rule.

The most interesting result of the experiment the disapproval of all the rules except one with the *founded implication* and also the *above average* quantifier. The fact leads to a conclusion that the p parameters of 4FT quantifiers are too restrictive, e.g. there should be 95% confidence of the rule when using *founded implication* quantifier.

4.3 Suitable Quantifiers' Settings

As the previous section showed, the default settings of a quantifier can be misleading. The next conducted experiment tries to find suitable quantifiers' settings, based on the *background knowledge* rule validation. We gradually decreased the p settings of the *founded implication* and *above average* quantifiers. We did not experiment with the *KL* quantifiers, because of the complexity of the problem[11]. With this technique, we could examine more *background knowledge* rules,

[11] The results need not to improve merely by changing a parameter of a quantifier. We also need to take the shape of the *KL* contingency table into consideration.

determine the value of the parameter for each rule and compute the average of the values for each examined dataset. New mining with the quantifier can be done with this average value and new relevant relationships in the data could be discovered.

Rule number	**FI**	**AA**
1	0.83	1,03
2	0.72	0.43
3	1	0.68
4	0.32	1.17
5	0.28	1.34
6	0.38	1.17
7	0.16	1.15
8	0.64	1.07

Table 4. Exact quantifiers values

As we can see in Table 4, the results of the experiment are rather disappointing for the *founded implication* quantifier. Majority of rules had the P value below 0.5. We got better results for the *above average* quantifier where the p parameter was only twice below 1. However, only once the value exceeded the desired 1.2 value.

4.4 Discussion

Considering the KL procedure, we obtained reasonable results for the *increasing dependence* and bad results for the *decreasing dependence* abstract quantifiers. This may be caused by the fact, that the *categorial attributes* R and C of the task setting contained few categories and thus irregularities of the KL contingency table (see [15] for details) could easily affect the quantifier.

Considering the 4FT procedure, there results for *above average dependence* were reasonable. On the other hand, the most used quantifier *founded implication* did not prove to be useful at all. This may be caused by the fact, that for rules no. 4, 5 and 6 (φ increasing, ψ decreasing) *founded implication* is not a suitable quantifier.

Although the formal theory of the quantifiers (KL and 4FT) is well developed [14], our experiments showed that semantically sound interpretation is yet to be researched. [9] is the first attempt of summarized semantic explanation of significant quantifiers.

5 Related Work

In [2], authors use background knowledge for subgroup discovery. Important part of the work tries to divide background knowledge into classes and deals separately with each class. Unfortunately the rules and formalization defined in this work does not belong to any of the classes defined.

In [5], authors developed ideologically similar approach: they used classical association mining in cooperation with a Bayesian network to store the knowledge from domain experts (here called *a priori expert knowledge*) and improved both the association rules mining and the Bayesian network in iterations. This approach is stronger from the methodological point of view (complex methodology is defined) and also enables revision of the domain expert knowledge. However, our background knowledge formalization is less restrictive than the Bayesian network and the GUHA procedures offer greater possibilities than the classical association mining.

[4, 11] show another formalization of background knowledge. It is based on qualitative models and used for induction learning. This model is not suitable for GUHA mining, mainly because the strict mathematical requirements of the model.

The data from STULONG itself have been matter of long run research [3, 10]. [16] deals with background knowledge rules annotation into a attribute matrix. This annotation is a simplification of our formalization without proper explanation of suggested *abstract quantifiers*.

6 Conclusion and Future Work

References

1. Agrawal R., Imielinski T., Swami A.: *Mining association rules between sets of items in large databases.* Proc. of the ACM SIGMOD Conference on Management of Data, p. 207 – 216
2. Atzmueller M., Puppe F.: *A Methodological View on Knowledge-Intensive Subgroup Discovery.* In: S. Staab and V. Svátek (Eds.): EKAW 2006, LNAI 4248, Springer-Verlag 2006, p. 318 – 325
3. Berka P., Tomečková M., Rauch J.: *The ECML/PKDD Discovery Challenge on the Atherosclerosis Risk Factors Data.* In: Popelínský L., Krátký M. (ed.): Znalosti 2005. Ostrava, VB TU Ostrava 2005, pp. 18-28. ISBN 80-248-0755-6.
4. Clark P., Matwin S.: *Using Qualitative Models to Guide Inductive Learning.* In: Proceedings 10th International Machine Learning Conference (ML93), p. 49–56
5. Fauré C., Delpart S., Boulicaut J., Mille A.: *Iterative Bayesian Network Implementation by Using Annotated Association Rules.* In: S. Staab and V. Svátek (Eds.): EKAW 2006, LNAI 4248, Springer-Verlag 2006, p. 326 – 333
6. Hájek P., Havel I., Chytil M.: *The GUHA method of automatic hypotheses determination.* Computing 1, 1966, p. 293 – 308
7. Hájek P., Havránek, T.: *Mechanising Hypothesis Formation - Mathematical Foundations for a General Theory.* Springer-Verlag: Berlin - Heidelberg - New York, 1978.
8. Kováč M., Kuchař T., Kuzmin A., Ralbovský M.: *Ferda, New Visual Environment for Data Mining.* Znalosti 2006, Conference on Data Mining, Hradec Králové 2006,p. 118 – 129 (in Czech)

9. Kupka D.: *User support 4ft-Miner procedure for Data Mining.* Master Thesis, Faculty of Mathematics and Physics, Charles University, Prague 2006 (in Czech)
10. Lucas N., Azé J.. Sebag M.: *Atherosclerosis Risk Identification and Visual Analysis.* In Berka, P. (ed.): Discovery Challenge Workshop Notes. ECML/PKDD-2002. Helsinki 2002.
11. Matwin S., Rouge T.: *Explainable Induction with an Imperfect Qualitative Model.* `http://citeseer.ist.psu.edu/matwin95explainable.html`
12. Ralbovský M.: *Usage of Domain Knowledge for Applications of GUHA Procedures*, Master Thesis, Faculty of Mathematics and Physics, Charles University, Prague 2006 (in Czech)
13. Ralbovský M., Kuchař T.: *Using Disjunctions in Association Mining.* In: Perner P.: Advances in Data Mining, Springer-Verlag 2007, to appear
14. Rauch J.: *Logic of Association Rules.* In: Applied Inteligence, Vol. 22, Issue 1, p. 9 – 28
15. Rauch J., Šimůnek M., Lín V.: *Mining for Patterns Based on Contingency Tables by KL-Miner First Experience.* In: LIN, Tsau Young, OHSUGA, Setsuo, LIAU, C. J., HU, Xiaohua (Eds.) Foundations and Novel Approaches in Data Mining. Berlin : Springer-Verlag, pp. 155–167. ISBN 3-540-28315-3
16. Rauch J., Tomečková M.: *System of Analytical Questions and Reports on Mining in Health Data – a Case Study.* Submitted to IADIS 2007
17. Řehák J., Řeháková B.: *Analysis of Categorized Data in Sociology* (in Czech). Academia: Prague, Czechoslovakia, 1986
18. Svátek V., Rauch J., Ralbovský M.: *Ontology-Enhanced Association Mining.* In: Ackermann, Berendt (eds.). Semantics, Web and Mining, Springer-Verlag, 2006