Using Disjunctions in Association Mining

Martin Ralbovský[1] and Tomáš Kuchař[2]

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

[2] Department of Software Engineering, Faculty of Mathematics and Physics
Charles University, Malostransk nm. 25, 118 01 Prague, Czech Republic
tomas.kuchar@gmail.com

Abstract. The paper focuses on usage of disjunction of items in association rules mining. We used the GUHA method instead of the traditional *apriori* algorithm and enhanced the former implementations of the method with ability of disjunctions setting between items. Experiments were conducted in our Ferda data mining environment on data from the medical domain. We found strong and meaningful association rules that could not be obtained without the usage of disjunction.

Keywords: Association Mining, Disjunction, GUHA Method, Ferda

1 Introduction

Association rules mining is an important technique widely used in the KDD community [8]. Most of the tools nowadays use the *apriori* algorithm, or its modifications [1] [2]. The algorithm searches for frequent (or large) itemsets with given minimal *support* and then calculates *confidence*. We will refer to this algorithm as to classical association mining. Its authors considered only the conjunctions (and possibly negations) of items.

Yet sometimes it is feasible to examine disjunctions of items. Consider following example: Medical expert wants to find associations between beer consumption and other characteristics of a patient (blood pressure, level of cholesterol, body mass index...). The examined data contains information about consumption of three different types of beer: light 7 degree beer, drought 10 degree beer and lager 12 degree beer[3]. It is likely to happen that the number of patients drinking 7 degree *or* 12 degree beer is higher then the number of patients drinking 7 degree *and* 12 degree beer. More formally, from the rule $A \to B$ one can get rule $A \to B \vee C$ easily than the rule $A \to B \wedge C$. For semantically close entities[4] one can therefore use disjunctions and mine for rules with higher support (and possibly other characteristics).

[3] This categorization of beer is traditional in the Czech Republic and represents the weight percentage of mash in the end product. 7 degree beer contains about 2% of alcohol, 10 degree beer about 3 to 4% and 12 degree beer about 4 to 5% of alcohol.

[4] Drinking of different types of beer is semantically close characteristics of a patient.

The aim of this paper is to present an enhancement of classical association mining with the possibility of disjunction setting between the items. One cannot use *apriori* for disjunctions, because the algorithm searches for frequent itemsets by binding items to already known itemsets (of length k) with conjunction to form itemsets (of length $k+1$). We used the older GUHA method instead, which mines for modifications of association rules. The generalization enables disjunctions between items and had several partial implementations before the personal computer era. We created a new implementation in our Ferda tool and conducted experiments with medical data using more strict requirements for rules than *support* and *confidence*. Meaningful rules have been found; these rules could not be found without disjunction usage and have interesting characteristics that should be subject of further research.

The paper is structured as follows: Section 2 explains the GUHA method and its relation to classical association mining. Section 3 states a brief history of tools implementing the GUHA method leading to our Ferda tool. Section 4 describes conducted experiments. Section 5 draws fields of further research and finally section 6 concludes the work.

2 Principles of association mining with GUHA

2.1 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 [4], [5]. For purpose of this paper let us explain only the basic principles of the method, as shown in Figure 1.

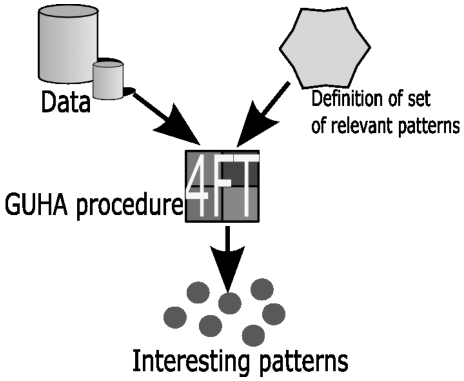


Fig. 1. The GUHA method

GUHA method is realized by GUHA procedures such as 4FT procedure to be described, located in the middle of the figure[5]. 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 2.2. The procedure automatically generates all the relevant patterns and verifies them against the provided data. Patterns that are true are output of the procedure.

Although GUHA is not in principle restricted to mining association rules, the most used GUHA procedures mine for generalized association rules, as defined in [12]. Section 2.3 introduces 4FT, procedure for association rules mining used in our work. Comparison study between the classical association mining and mining using GUHA can be found in [6].

2.2 Definition of set of relevant patterns

This section shows how set of relevant patterns is defined in association rules mining with GUHA. We use the term attribute in the sense of *categorial attribute*, i.e. attribute with finite number of values.

Definition 1 *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**.

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**.

The above stated definition was introduced in [12] when formalizing association rules. *Boolean attributes* are used as antecedents or succedents[6] in GUHA procedures, as will be described in section 2.3. Our Ferda tool is the first program to enable full *Boolean attribute* definition including disjunction and recursion. Example 1 shows us creation of *Boolean attributes* from the beer consumption example from the introduction.

Example 1
The examined data includes three attributes: **beer7= {no, low, high}**, **beer10= {no, low, high}** and **beer12= {no, low, high}** for consumption of 7, 10 and 12 degree beer respectively.

Examples of *basic Boolean attributes* are **beer7(no)**, **beer10(no, low)** or **beer12(high)**[7].

Then we combine *basic Boolean attributes* with logical operators to form a rule:

[5] Even though we present only one GUHA procedure in this work, there are five more procedures working above one data table implemented in Ferda and also two relational procedures under development.

[6] In classical association mining called consequents.

[7] Obviously, not all the subsets of an attribute are meaningful to verify. Our method allows user to define special subsets such as subsets with a given length, intervals, cyclic intervals or cuts for ordinal data.

(beer7(no) $\vee$ **beer10(no, low))** $\wedge$ $\neg$**beer12(high)**,
which is an example of *Boolean attribute*.

2.3 4FT procedure

Classical 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 rules in form $\varphi \approx \psi$, where φ and ψ are *Boolean attributes* and $\approx$ is a *4ft-quantifier*[8]. Relation $\varphi \approx \psi$ is evaluated on the basis of *4ft table*, as shown in Table 1.

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*. By the expression **strict quantifier** we mean that there are no rules that satisfy the quantifier in the usual case. Occurrence of such quantifier means a very strong relation in the data. In the following sections we present three quantifiers used in our work, the *founded implication*, *double founded implication* and *founded equivalence* quantifiers[9]. This part of the paper was greatly inspired by [11], where detailed explanation of the most used quantifiers can be found.

2.4 Founded Implication Quantifier

The founded implication is basic quantifier for the 4FT procedure introduced in [4] and is defined by following condition:

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

[8] 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.

[9] There are many other quantifiers invented and implemented for the 4FT procedure.

Here $Base$ 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, $\frac{a}{a+b+c+d}$. So, the $Base$ parameter corresponds to the *support* and p to the *confidence* parameters of classical association mining. When using the 4FT procedure with *founded implication quantifier* and constructing *Boolean attributes* only with conjunctions, we get the same results as if using classical association mining. Quantifier can be verbally interpreted with the expression *tendency to.*

Example 2
Association rule *Patients that drink 12 degree beer tend be overweight* is an example of rule we can found with *founded implication.* This rule can be formally written as **beer12(high)** $\Rightarrow_{p,Base}$ **BMI(overweight)**, where $\Rightarrow_{p,Base}$ stands for *founded implication.*

2.5 Double Founded Implication Quantifier

The *double founded implication* quantifier enriches the *founded implication* with symmetry feature. Symmetry means that when $\varphi \approx \psi$ is valid, when $\psi \approx \varphi$ should be also valid. The quantifier has also threshold parameters $Base$ and p and is defined by following condition:

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

Again, we will use the relative representation of $Base$, $\frac{a}{a+b+c+d}$. We consider *double founded implication* a *strict quantifier.* However, we wanted to use the quantifier in our experiments to question the possibilities of disjunctions. The most suitable verbal expression for the quantifier is *relation of equivalence.*

Example 3
The sign for *double founded implication* quantifier is $\Leftrightarrow_{p,Base}$. The rule **beer12(high)** $\Leftrightarrow_{p,Base}$ **BMI(overweight)** with the *Boolean attributes* from example 2 can be verbally interpreted as *Drinking 12 degree beer is in relation of equivalence with being overweight among the observed patients.*

2.6 Founded Equivalence Quantifier

The last presented quantifier is the *founded equivalence.* It is a stronger quantifier than *founded implication* in terms of equivalence; ability of two entities to attain the same logical values. The condition for the quantifier is

$$a \leq Base \wedge \tfrac{a+d}{a+b+c+d} \geq p$$

For *founded equivalence*, we will also use the relative representation of $Base$, $\frac{a}{a+b+c+d}$. The fraction $\frac{a+d}{a+b+c+d}$ means the proportion of objects in the data matrix having φ and ψ both equal to 0 or 1, to all objects. Similarly, $Base$ and p are threshold parameters for the quantifier. As well as *double founded implication*, the *founded equivalence* is considered to be a *strict quantifier*. Quantifier can be verbally interpreted as *equivalent occurrence*.

Example 4
The sign for the *founded equivalence* quantifier is $\equiv_{p,Base}$. Generalized association rule **beer12(high)** $\equiv_{p,Base}$ **BMI(overweight)** can be translated to verbal form as *Consumption of 12 degree beer and being overweight has equivalent occurrence among the observed patients*.

3 GUHA and Ferda

We would not achieve results presented in this work without 40 years long research of the GUHA method and development of tools that implemented individual GUHA procedures. This section acknowledges achievements made by researchers and developers in the past and briefly describes history that lead to the state-of-the-art Ferda tool. See [3] for more information about history of GUHA method.

The development of first GUHA procedure started in 1956. In modern terminology, it mined for association rules with given *confidence* with one item as a antecedent and one item as a consequent [3]. The results, published in [4], were clearly ahead of their time, long before terms like data mining or knowledge discovery from databases were invented.

First GUHA procedure to consider disjunctions was the IMPL procedure introduced in [5]. The procedure mined for rules in form *CONJ* $\Rightarrow$ *DISJ*, where *CONJ* and *DISJ* are elementary conjunctions and disjunctions[10]. $\Rightarrow$ is an *implicational quantifier*[11]. The implementation of the procedure [13] [15] used for the first time the bit string approach[12]. The input data were represented by strings of bits, which dramatically increased performance of the procedure.

The LISp-Miner tool[13] started in 1996 and contributed greatly to the level of contemporary GUHA tools by implementing six GUHA procedures and implementing coefficients - generation of special types of subsets of an attribute [14]. GUHA procedure 4ft-Miner implemented in LISp-Miner is predecessor of procedure 4FT introduced in this work. 4ft-Miner does not allow construction of *Boolean attributes*, it constructs *partial cedents* instead and until very recently it did not allow disjunctions. *Partial cedent* is a restricted non recursive *Boolean attribute*, more details are to be found in [14].

Ferda started as a student project to create a new visual environment for the LISp-Miner system [9]. In the first version, creators used the LISp-Miner GUHA

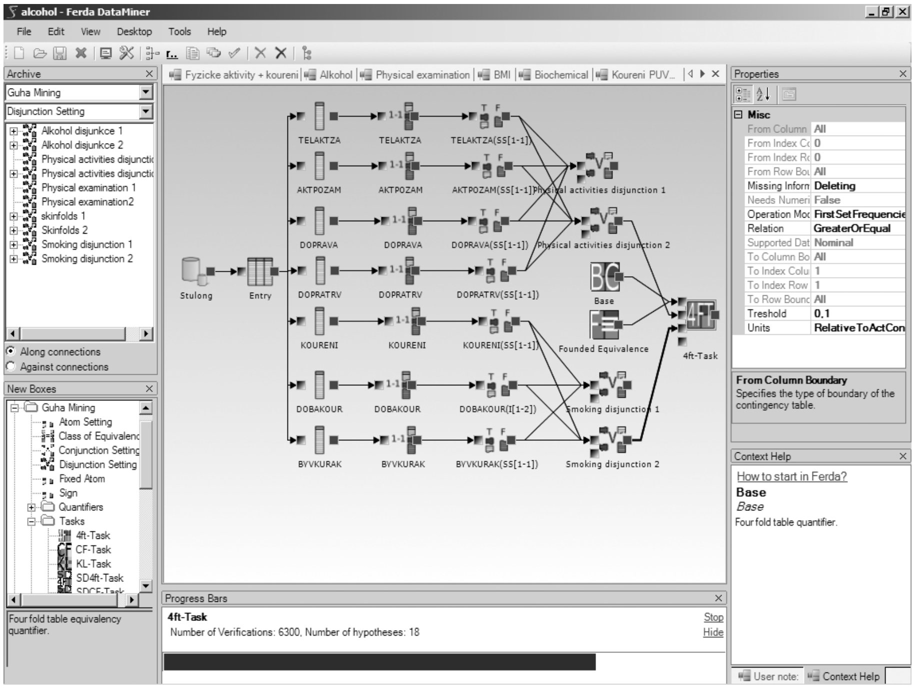
[10] Elementary conjunction is a conjunction made from one element subsets of an attribute.

[11] There are formal classes of *4ft-quantifiers* defined in [12]. *Implicational quantifiers* are one of the classes.

[12] Also known as *granular computing*.

[13] See `http://lispminer.vse.cz`.

procedures. The second version of the system, implemented in work [10], uses the bit string approach and enables full definition of *Boolean attribute*. It is the first modern tool (runs on personal computers) to implement disjunctions and recursion of *basic Boolean attributes*. The procedure 4FT implemented in Ferda is the most generalized version of the original ASSOC procedure defined in [5] The user environment is shown in Figure 2.



Four fold table equivalency quantifier.

From Column	All
From Index Cc	0
From Index Rc	0
From Row Bou	All
Missing Inforn	Deleting
Needs Numeri	False
Operation Moc	First Set Frequencie
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.

4ft-Task

Number of Verifications: 6300, Number of hypotheses: 18

Fig. 2. Ferda environment

4 Experiments

In order to test the possibilities of disjunctions of items, we carried out experiment, which consisted of testing number of analytical questions. We chose the STULONG data set, introduced in section 4.1. The limitations and criteria that led to the experiment setup are described in sections 4.2 and 4.3. Performance is discussed in section 4.4. We found interesting and also some unexpected results which are summarized in section 4.5.

4.1 STULONG Data Set

We decided to use the STULONG medical data set for our experiments. The data set contains data about longitudinal study of atherosclerosis risk factors. There are two main reasons to choose this data set. First reason is that STULONG is relativelly known among KDD researchers – it served as the examined data set for three ECML/PKDD Discovery challenges. There are meaningful analytical questions defined on the data set to be examined, which is the second reason. In our experiments we wanted to answer these questions[14].

4.2 Limitations

Before explaining setup of the experiments, let us note two major limitations of mining disjunctions in general. These limitations affect our experiments as well. First limitation is generation of *non prime* rules. The rule is *prime* when it is true in the examined data and when the rule cannot be derived from a more simple rule also true in the examined data. More details can be found in [5]

Our implementation does not guarantee generation of prime rules. Therefore we expect the number of valid rules to rise dramatically when using disjunctions with quantifiers that are not *strict*. However, we may use disjunctions with *strict quantifiers*, where it is a common case that no rules are found at all.

The other limitation is interpretation of rules with disjunction. The motivation example of beer consumption in section 1 showed that it makes sense to use disjunctions with semantically close attributes, possibly synonyms or taxonomically bound attributes. Interpretation of disjunction of random attributes is rather problematic. Therefore we used for disjunctions only the attributes of the same attribute groups[15].

[14] EUROMISE: The STULONG Project `http://euromise.vse.cz/stulong`
The STULONG Project is partially supported by project no.LN00B107 of the Ministry of Education of the Czech Republic and by grant no.2003/23 of the Internal Grant Agency of the University of Economics, Prague. The STULONG study was carried out at the 2nd Department of Medicine, 1st Faculty of Medicine of Charles University and Charles University Hospital (head Prof. M. Aschermann, MD, SDr, FESC), under the supervision of Prof. F. Boudík, MD, ScD, with collaboration of M. Tomečková, MD, PhD and Ass. Prof. J. Bultas, MD, PhD. The data were transferred to electronic form by the European Centre of Medical Informatics, Statistics and Epidemiology of Charles University and Academy of Sciences (head Prof. RNDr. J. Zvárová, DrSc).

[15] Groups of attributes, i.e. *physical examination* are defined in the STULONG data set.

4.3 Setup

From above stated limitations we concluded a setup for experiments. We answered 15 analytical questions concerning relations between significant characteristics of patients' entry examination[16]:

1. *What are the relations between social factors and the following characteristics of men in the respective groups:*
   (a) *Physical activity at work and in free time*
   (b) *Smoking*
   (c) *Alcohol consumption*
   (d) *BMI*
   (e) *Blood pressure*
   (f) *Level of total cholesterol, HDL cholesterol, triglycerides*
2. *What are the relations between physical activity at work and in free time and the following characteristics of men in the respective groups:*
   (a) *Smoking*
   (b) *Alcohol consumption*
   (c) *BMI*
   (d) *Blood pressure*
   (e) *Level of total cholesterol, HDL cholesterol, triglycerides*
3. *What are the relations between alcohol consumption and the following characteristics of men in the respective groups:*
   (a) *Smoking*
   (b) *BMI*
   (c) *Blood pressure*
   (d) *Level of total cholesterol, HDL cholesterol, triglycerides*

The experiment consisted of two steps. The first tried to answer the questions without usage of disjunctions. The second step used the task settings from the first step and allowed disjunctions of length 2. We applied the *double founded implication* quantifier with settings *p=0.9*, *Base=0.1* and the *founded equivalence* quantifier with settings *p=0.9* and *Base=0.1*. Below stated are the goals of the experiment:

1. Show the difference between using and not using disjunctions.
2. Use disjunctions with *strict quantifiers*.
3. Find interesting rules that contain disjunction.

[16] The analytical questions can be found at `http://euromise.vse.cz/challenge2004/tasks.html`

4.4 Performance

It is shown in [14] that the 4FT procedure operation time without disjunctions is approximately linear to number of rows of the data matrix. Moreover, the bit string approach takes advantage of fast instructions in the processor, which makes running times acceptable for most tasks. When using disjunctions, the operation time rises due to the fact, that search space is not reduced by adding conjunctions decreasing *support* of the rule. However, practical experience shows that there is no need to be concerned, because the running times are acceptable.

In our experiment we used a Pentium M 1,7GHz processor with 1 GB of RAM and Windows XP. We noted the running times of tasks designed to answer proposed analytical questions. Without disjunctions, minimal running time was 0.310 seconds, maximal running time was 2.543 seconds and average running time was 0.829 seconds[17]. When using disjunctions, minimal running time was 0.370 seconds, maximal running time was 345.997 and average running time was 38.9 seconds. In the maximum case, procedure constructed and verified almost 8 million contingency tables.

4.5 Results

After conducting the first step of the experiment, we found 0 rules for 14 of 15 analytical questions and one rule for question 3.(b). This result confirmed our presumption, that *double founded implication* and *founded equivalence* are *strict quantifiers.* When using disjunctions, we found minimum 1 and maximum 185 rules per analytical question. Although we agree that number of rules found is not a good metrics of measuring performance of new data mining technique, we think that the shift from zero rules found to non-zero rules found is significant.

Let us consider the significance of the rules. In order to reduce the amount of rules presented, we consider for simplicity only the analytical question 3.(b). We may limit our analysis, because all the rules found show similar characteristics. Possible rules answering the question were presented as examples throughout the article. Moreover, a rule for this analytical question was found during step one of the experiment.

$$Beer7(No) \Leftrightarrow_{p=0.986, Base=0.968} BMI(Normalweight, Overweight)$$

is the rule found without disjunction usage. It can be explained by the fact that 7 degree beer is very rare and it was mainly used for hydration of manual workers in extremely hot working environment (glassmakers or metallurgists). Therefore majority of population did not drink this type of beer.

Table 2 shows rules found with disjunction usage. The rules were valid both for *double founded implication* (DFI) and *founded equivalence*(FE) quantifiers. Numbers in these columns represent actual values of quantifiers, $\frac{a}{a+b+c}$ for *double*

[17] Performance tests between the 4FT procedure implemented in Ferda and LISp-Miner can be found in [10].

Antecedent	**Succedent**	**DFI**	**FE**	**Base**
Beer10(No) ∨ Beer12(No)	BMI(Normal weight, Overweight)	0.929	0.932	0.931
Beer12(No) ∨ Wine(Yes)	BMI(Normal weight, Overweight)	0.906	0.909	0.909
Beer12(No) ∨ Liquor(Yes)	BMI(Normal weight, Overweight)	0.905	0.908	0.909
Beer12(No) ∨ Alcohol(Occasionally)	BMI(Normal weight, Overweight)	0.902	0.904	0.904
Beer12(No) ∨ BeerDaily(<1 liter)	BMI(Normal weight, Overweight)	0.946	0.948	0.948
Beer12(No) ∨ WineDaily($< \frac{1}{2}$ liter)	BMI(Normal weight, Overweight)	0.9	0.903	0.903
Wine(No) ∨ WineDaily($< \frac{1}{2}$ liter)	BMI(Normal weight, Overweight)	0.951	0.951	0.951
Liquor(No) ∨ LiquorDaily(<1 dL)	BMI(Normal weight, Overweight)	0.923	0.924	0.924

Table 2. Rules found

founded implication, $\frac{a+d}{a+b+c+d}$ for *founded equivalence* and $\frac{a}{a+b+c+d}$ for *Base*. According to high values of both quantifiers and also to the high support of the rules (the *Base* parameter), we have found very strong rules containing disjunctions. Despite the initial concerns about interpretability of rules with disjunctions, the rules can be easily interpreted and comprehended. We are aware of the fact, that the rules do not show any surprising relations[18]. Yet they show strong relations in data, where almost no relations without disjunction usage was discovered.

To conclude the experiment, all three goals from section 4.3 were reached. We showed the difference of mining with and without disjunctions by getting almost none rules without disjunctions and more rules with disjunctions. We managed to use *strict quantifiers* and we also found interesting rules containing disjunctions. The experiment also raised many questions about further development, some of which will be discussed in the following section 5.

5 Further research

There are several directions to improve disjunctions using in association mining. This section explains the directions in more detail. The first direction is optimalization of disjunctions verifications. We showed in section 4.4 that average running times for average size tasks is acceptable. However there is still room for optimizing. As was stated before, one cannot apply pruning used with conjunctions. Solution can lie in ordering of *basic Boolean attributes* according to their support. However the ordering itself can be a significant performance problem.

Another direction is to implement generation of prime rules only. The theory about prime rules is explained in [5], [12] includes information about deduction rules, which can be used when checking prime property of an association rules.

[18] This may be a problem of mining association rules in general.

When evaluating results of our experiments, we came acros an interesting coincidence between values of quantifiers, mainly the *Base* and p values of *double founded implication* and *founded equivalence* quantifiers. A natural question occurs whether this is only a coincidence or if it is some kind of functional dependence. The quantifiers are known for years, yet without disjunctions we were not able on any data to mine a reasonable amount of rules to show the coincidence[19]. Examination of quantifiers as functions $f : \Re^4 \rightarrow \Re$ by finding functional extremes with the aid of calculus is another direction of further research, which would answer the question of coincidence or dependence between quantifiers.

Boolean attribute was presented in the article. The attribute can reach values *false* or *true* (0 or 1). *Fuzzy attribute* can also be defined, reaching values from the interval $\langle 0, 1 \rangle$. Then fuzzy *4ft tables* can be constructed and fuzzy quantifiers can be defined[20]. [7] is the inspiration for this direction.

Last, but not least direction of further research is cooperation with domain experts to evaluate usability of rules with disjunctions. We mined over medical data and presented association rules comprehensible to non medical experts. Presenting the rules found to the medical experts provides valuable feedback for us.

6 Conclusion

We present an enhancement of association mining with the possibility of setting disjunctions between the items. The classical *apriori* algorithm was not suitable for disjunctions. Instead older GUHA method was applied. The 4FT procedure is used, which mines for rules in form $\varphi \approx \psi$ where φ and ψ are *Boolean attributes* and $\approx$ is a *4ft-quantifier*. *Boolean attribute* is a recursive structure, where disjunction can be used. 4FT procedure was implemented in our Ferda data mining tool.

Experiments were conducted to show the usability of disjunctions in association mining. We tried to answer number of analytical questions from the STULONG medical data set containing statistical information of atherosclerosis risk factors. We applied *double founded implication* and *founded equivalence* quantifiers, which are considered to be *strict*, that is to return no rules in most cases. The experiments showed difference between mining with and without disjunctions and found strong interpretable rules containing disjunctions in the data. The experiments also showed some issues, which should be subjects of further research.

Acknowledgements

This work was supported by the project MSM6138439910 of the Ministry of Education of the Czech Republic, project IG407056 of University of Economics,

[19] This corresponds to considering the quantifiers as *strict*.

[20] Naturally, some of the existing quantifiers could not be used.

Prague and by the project 201/05/0325 of the Czech Science Foundation. We acknowledge the contribution of our research colleagues Jan Rauch and Daniel Kupka for valuable comments and reviews.

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. Agrawal R., Mannila H., Srikant R., Toivonen H., Verkamo A.: *Fast discovery of association rules.* Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R., eds.: Advances in Knowledge Discovery and Data Mining. AAAI Press, Menlo Park (1996) p. 307 – 328
3. Hájek P.: *The GUHA Method in the Last Century and Now.* Znalosti 2006, Conference on Data Mining, Brno 2006, p. 10 – 20 (in Czech)
4. Hájek P., Havel I., Chytil M.: *The GUHA method of automatic hypotheses determination.* Computing 1, 1966, p. 293 – 308
5. Hájek P., Havránek, T.: *Mechanising Hypothesis Formation - Mathematical Foundations for a General Theory.* Springer-Verlag: Berlin - Heidelberg - New York, 1978.
6. Hájek P., Holeňa M.: *Formal logics of discovery and hypothesis formation by machine.* Theoretical Computer Science 292 (2003) p. 345 – 357
7. Holeňa M.: *Fuzzy hypotheses testing in framework of fuzzy logic.* Fuzzy Sets and Systems, 149, p. 229 – 252
8. KDNuggets Polls, *Data mining/analytic techniques you use frequently.* `www.kdnuggets.com/polls/2005/data_mining_techniques.htm`
9. 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)
10. Kuchař T.: *Experimental GUHA Procedures*, Master Thesis, Faculty of Mathematics and Physics, Charles University, Prague 2006 (in Czech)
11. Kupka D.: *User support 4ft-Miner procedure for Data Mining.* Master Thesis, Faculty of Mathematics and Physics, Charles University, Prague 2006 (in Czech)
12. Rauch J.: *Logic of Association Rules.* In: Applied Inteligence, Vol. 22, Issue 1, p. 9 – 28
13. Rauch J.: *Some Remarks on Computer Realisations of GUHA Procedures.* International Journal of Man-Machine Studies 10, 1978, p. 23 – 28
14. Rauch J., Šimůnek, M.: *An Alternative Approach to Mining Association Rules* Lin T Y, Ohsuga S, Liau C J, and Tsumoto S (eds): Foundations of Data Mining and Knowledge Discovery, Springer-Verlag, 2005 p. 219 – 239
15. Rauch J., Šimůnek, M.: *GUHA Method and Granular Computing.* Proceedings of IEEE International Conference on Granular Computing, 2005, `http://www.cs.sjsu.edu/~grc/grc2005/index.html`