Modifying Logic of Discovery for Dealing with Domain Knowledge in Data Mining

Jan Rauch

Faculty of Informatics and Statistics, University of Economics, Prague *
nám W. Churchilla 4, 130 67 Prague, Czech Republic rauch@vse.cz

Abstract. Logic of discovery was developed in 1970's as an answer to questions "Can computers formulate and justify scientific hypotheses?" and "Can they comprehend empirical data and process it rationally, using the apparatus of modern mathematical logic and statistics to try to produce a rational image of the observed empirical world?". Logic of discovery is based on observational and theoretical languages and on inductive inference corresponding to statistical approaches. Formulas of observational language concern analyzed observational data and formulas of theoretical language concern suitable state dependent structures. The goal of the paper is to discuss a possibility to adapt the logic of discovery to data mining.

1 Introduction

Logic of discovery is developed in the book [2] which starts with questions:(Q_1) – *Can computers formulate and justify* scientific *hypotheses?* (Q_2) – *Can they comprehend empirical data and process it rationally, using the apparatus of modern mathematical logic and statistics to try to produce a rational image of the observed empirical world?* Answers are based on a scheme of inductive inference:

$$\frac{\text{theoretical assumptions, observational statement (s)}}{\text{theoretical statement}} .$$

This schema means that having *accepted* theoretical assumptions and having *verified* observational statements concerning analyzed data, we *accept* a theoretical statement. The schema leads to additional questions L0 - L4: (L0) In what languages does one formulate observational and theoretical statements? (L1) What are rational inductive inference rules bridging the gap between observational and theoretical sentences? (L2) Are there rational methods for deciding whether a theoretical statement is justified? (L3) What are the conditions for a theoretical statement or a set of theoretical statements to be of interest with respect to the task of scientific cognition? (L4) Are there methods for suggesting such a set of statements, which is as interesting (important) as possible?

* This paper was prepared with the support of *Institutional funds for support of a long-term development of science and research at the Faculty of Informatics and Statistics of University of Economics, Prague.*

Answering questions (LO) – (L2) leads to logic of induction, answers to questions (L3) and (L4) lead to logic of suggestions. Answers to questions (LO) – (L4) constitute a logic of discovery. Very detailed answers to questions L0 - L4 are given in the book [2] and the logic of discovery is developed. Observational and theoretical calculi are developed as languages for observational and theoretical statements respectively. Principles of logic of discovery are briefly outlined in section 2.

Various observational calculi are defined in [2]. The most studied calculi are monadic observational predicate calculi. They can be understood as a modification of classical predicate calculi – only finite models are allowed and generalized quantifiers are added. Finite models correspond to observational data we analyze and generalized quantifiers make possible to express important statements on observational data. Monadic observational predicate calculi were modified and significantly simplified in [5] such that they can be understood as a logic of association rules. Association rules - formulas of these calculi are more general than association rules defined in [1]. These generalized association rules are produced by the procedure 4ft-Miner [7].

Application of domain knowledge in data mining is introduced among 10 challenging problems of data mining [4], see also `http://www.cs.uvm.edu/~icdm/`. Domain knowledge is also referred as background knowledge, world knowledge, or business knowledge. Results presented in [5] make possible to deal with domain knowledge in the process of data mining. A way of filtering out consequences of domain knowledge from results of the 4ft-Miner procedure is outlined in [6]. It is based on application of logic of association rules [5]. The goal of this paper is to present (in a given scope) a theoretical elaboration of this approach.

We are going to modify logic of discovery developed in [2]. A modified theoretical language is intended to express items of domain knowledge intuitively understandable to domain experts without experience in data mining. Formulas of observational language correspond to patterns produced by data mining procedures. There is a new way of a correspondence between theoretical and observational languages.

A set of atomic consequences is assigned to each item of domain knowledge (i.e. to a formula of the theoretical language). The atomic consequences are simple formulas of observational calculus such that the assignment can be done by a domain expert. Deduction rules among observational formulas are then used to spread the consequences of items of domain knowledge among additional formulas of observational calculus.

Let us emphasize that the theoretical language expressing domain knowledge totally differs from the observational language of patterns produced by data mining procedures. The correspondence between these languages is ensured by the atomic consequences defined by the domain expert and by deduction rules in the observational calculus. First we outline general features of this approach and then we elaborate it for association rules. We call resulting modification of logic of discovery as *logic of mining of association rules*. No analogous approach concerning association rules is known to the author.

Logic of discovery is sketched in section 2. Principles of its modification are discussed in section 3. Modified observational and theoretical languages are introduced in sections 4 and 5. System 4ft-Discoverer integrating both introduced theoretical principles and software procedures for dealing with domain knowledge is discussed in section 6.

2 Logic of Discovery

Semantic system $\mathcal{S} = \langle Sent, \mathcal{M}, V, Val \rangle$ is determined by a non-empty set *Sent* of *sentences*, a non-empty set $\mathcal{M}$ of *models*, a non-empty set V of *abstract values* and an *evaluating function* $Val : (Sent \times \mathrm{M}) \rightarrow V$ [2]. If $\varphi \in Sent$ and $\underline{M} \in \mathcal{M}$ then $Val(\varphi, \underline{M})$ is the value of φ in $\underline{M}$. Semantic system $\mathcal{S} = \langle Sent, \mathcal{M}, V, Val \rangle$ is *observational* if *Sent*, $\mathcal{M}$, V are recursive sets and Val is a partial recursive function [2]. Observational semantic system $\mathcal{S}^O = \langle Sent^O, \mathcal{M}^O, V^O, Val^O \rangle$ corresponding to analyzed data and theoretical semantic system $\mathcal{S}^T = \langle Sent^T, \mathcal{U}^T, V^T, Val^T \rangle$ corresponding to the whole set of objects, we are interested in are developed in [2].

Observational predicate calculus is a result of modifications of predicate calculi - only finite models are allowed and generalized quantifiers are added [2], see introduction. System of closed formulas of such calculus is an observational semantic system. Observational predicate calculus with formulas corresponding to association rules was developed in [2]. Question of rationality of inductive inference rule is very important. It is studied in [2] using statistical approaches. It leads to observational predicate calculi with generalized quantifiers corresponding to statistical hypothesis tests and to theoretical semantic system $\mathcal{S}^T = \langle Sent^T, \mathcal{U}^T, V^T, Val^T \rangle$ with state dependent structures. However, the more detailed description is out of the scope of this paper.

Using induction rules based on statistical methods usually means there is 1:1 correspondence between observational and theoretical statements. Thus a task of a suggestion of interesting theoretical statements can be converted to a task of a suggestion of interesting observational statements. The GUHA method is defined in [2] to solve this task. The method is carried out using GUHA procedures. A GUHA procedure is a computer program the input of which consists of analyzed data and a set of parameters defining the large set of relevant observational patterns. Its output is a set of all prime patterns. A pattern is prime if it is true in the analyzed data, and if it does not logically follow from another output simples pattern.

The most used GUHA procedure is the ASSOC procedure. It deals with enhanced association rules. It was several times implemented and many times applied, see e.g. [3]. One of its implementations is the procedure 4ft-Miner, see introduction and section 6.1. Implementations of the ASSOC procedure use theoretical results (namely deduction rules) concerning observational predicate calculi [2]. Logic of association rules involves additional both theoretically interesting and practically important results [5, 6]. Meaning of the GUHA method and thus also meaning of logic of discovery for data mining is summarized in [3].

3 Modifying Logic of Discovery

3.1 Starting points

Our starting points are: (1) Data we are dealing with do not satisfy requirements for application of statistical approaches. An example of such data is data about patients of a particular hospital. (2) Objects described by our data belong to a broader set of objects. An example of such a broader set is a set of residents in a region to which the hospital belongs. (3) We have various items of knowledge related to a particular data set. An example is identification of a particular device used to measure each of the observed patients. (4) We have various items of knowledge related to the broader set of objects that are not directly recorded for each object. An example is information on specific vaccination applied in a region in question. (5) We have various items of general knowledge about attributes of objects described in our data. An example is a commonly accepted fact that if weight increases, then blood pressure increases too. (6) We have data mining procedures, which are able to produce a lot of strong patterns valid in given data. Examples are the apriori algorithm [1] and the procedure 4ft-Miner, see section 6.1. Both produce association rules.

(7) Most of the patterns produced by the data mining procedure are uninteresting because of they are consequences of the above mentioned items of knowledge. (8) There are groups of patterns hidden in patterns produced by the data mining procedure such that each of the groups can be considered as a consequence of a yet not known item of knowledge.

Our goal is to modify logic of discovery such that we will be able to: (I) use items of knowledge mentioned in points 3 - 5; (II) filter out consequences of the above introduced items of knowledge from results of mining procedures, see point 7; (III) recognize a group of patterns, which can be considered as a consequence of a (yet not known) item of knowledge, see point 8.

We assume to use GUHA procedures as data mining procedures together with results on a related observational logical semantic system. To achieve requirements (I)–(III) we are going to: (A) Enhance observational semantic system by features making possible to capture items of knowledge related to particular data sets, see (3) above. (B) Enhance theoretical semantic system by features making possible to capture both items of general knowledge and items of knowledge related to the broader set, see (4) and (5) above. (C) Enhance theoretical semantic system by function $Cons$ assigning to each item $\mathcal{I}$ of knowledge according to (B) sets of formulas $Cons(\mathcal{I})$ of a corresponding observational system (i.e. a set of patterns produced by a GUHA data mining procedure). Set $Cons(\mathcal{I})$ is assumed to be a *set of atomic consequences of* $\mathcal{I}$. (D) To develop a new analytical procedure G-FILTER for each GUHA procedure. G-FILTER will filter out all consequences of given items of knowledge from output of the GUHA procedure. This way requirement II) will be achieved. (E) To develop a new analytical procedure G-SYNT for each GUHA procedure. G-SYNT will recognize groups of patterns, which can be considered as a consequence of a (yet not known) items of knowledge. This way requirement III) will be achieved.

We present principles of modification of logic of discovery using results related to logic of association rules [5]. We deal with data matrices introduced in section 3.2. The principles of modification of observational and theoretical systems are outlined in sections 3.3 and 3.4. The procedures G-FILTER and G-SYNT are presented by description of procedures 4ft-Filter and 4ft-Synt, which are related to the procedure 4ft-Miner, see section 6.2.

3.2 Data Matrices

We consider data matrices with values – natural numbers only. The natural numbers represent categories, i.e. possible values of observed attributes $A_1, \ldots, A_K$. Columns of the data matrix correspond to attributes. Rows correspond to observed objects, e.g. patients. An example of the data matrix is in the left part of figure 1.

object	A_1	...	A_K
o_1	1	...	6
$\vdots$	$\vdots$	$\ddots$	$\vdots$
o_n	1	...	1

Data matrix - informal view

object	f_1	...	f_K
o_1	$f_1(o_1)$	...	$f_K(o_1)$
$\vdots$	$\vdots$	$\ddots$	$\vdots$
o_n	$f_1(o_n)$	...	$f_K(o_n)$

Data matrix $\mathcal{M} = \langle M, f_1, \ldots, f_K \rangle$

Fig. 1. Data matrix

There is only the finite number of categories for each attribute. Let us assume that the number of categories in a column is t and that the categories are natural numbers $1, \ldots, t$. All values in the data matrix are then described by the numbers of categories for each column. The whole information on the number of columns and categories in the data matrix is then given by type of data matrix: A *type of data matrix* is a K-tuple $\mathcal{T} = \langle t_1, \ldots, t_K \rangle$ where $t_i \geq 2$ are natural numbers for $i = 1, \ldots, K$.

We use a more formal definition of a data matrix with the number of columns and the numbers of possible values in particular columns given by the type $\mathcal{T} = \langle t_1, \ldots, t_K \rangle$: A *data matrix of the type* $\mathcal{T}$ is a $K+1$-tuple $\mathcal{M} = \langle M, f_1, \ldots, f_K \rangle$, where M is a non-empty finite set and f_i is the unary function from M to $\{1, \ldots, t_i\}$ for $i = 1, \ldots, K$. Set M is a *set of rows* of data matrix $\mathcal{M}$. Set M is called a *domain* of data matrix $\mathcal{M}$. We write $M = Dom(\mathcal{M})$. An example of data matrix $\mathcal{M} = \langle M, f_1, \ldots, f_K \rangle$ is in the right part of figure 1. We assume that $M = \{o_1, \ldots, o_n\}$.

3.3 Modifying Observational Semantic System

Observational semantic system $\mathcal{S}^{\mathcal{T}} = \langle \mathtt{M}^{\mathcal{T}}, \mathcal{L}_{\mathcal{C}}, Val_{\mathcal{C}}, \mathcal{L}_{\mathtt{M}^{\mathcal{T}}}, Val_{\mathtt{M}^{\mathcal{T}}} \rangle$ is used instead of $\mathcal{S}^{O} = \langle Sent^{O}, \mathcal{M}^{O}, V^{O}, Val^{O} \rangle$ introduced in section 2. Set $\mathtt{M}^{\mathcal{T}}$ of all data

matrices $\mathcal{M}$ of type $\mathcal{T}$ is used instead of $\mathcal{M}^O$ and two languages – $\mathcal{L}_\mathcal{C}$ and $\mathcal{L}_{\mathtt{M}^\mathcal{T}}$ are used instead of $Sent^O$. We always use set {0,1} (i.e. {*false*, *true*}) as a set of possible values of formulas of our languages instead of V^O.

$\mathcal{L}_\mathcal{C}$ is a language of a logical calculus formulas of which correspond to patterns produced by a data mining procedure (i.e. GUHA procedure in our case). There is an evaluation function $Val_\mathcal{C} : (\mathcal{L}_\mathcal{C} \times \mathtt{M}^\mathcal{T}) \rightarrow \{0,1\}$. If $\varphi \in \mathcal{L}_\mathcal{C}$ and $\mathcal{M} \in \mathtt{M}^\mathcal{T}$ then $Val_\mathcal{C}(\varphi, \mathcal{M})$ is the value of φ in $\mathcal{M}$. If it is $Val_\mathcal{C}(\varphi, \mathcal{M}) = 1$ then φ *is true* in $\mathcal{M}$, otherwise φ *is false* in $\mathcal{M}$.

$\mathcal{L}_{\mathtt{M}^\mathcal{T}}$ is a language intended to express items of knowledge related to particular data matrices, see point (3) in section 3.1. It is a set $\Theta = \{\theta_1, \ldots \theta_R\}$ of formulas corresponding to features of $\mathcal{M}$, each of them can be true or false. There is an evaluation function $Val_{\mathtt{M}^\mathcal{T}} : (\Theta \times \mathtt{M}^\mathcal{T}) \rightarrow \{0,1\}$. If it is $\theta \in \Theta$ and $\mathcal{M} \in \mathtt{M}^\mathcal{T}$ then $Val_{\mathtt{M}^\mathcal{T}}(\theta, \mathcal{M})$ is the value of feature θ for $\mathcal{M}$. If it is $Val_{\mathtt{M}^\mathcal{T}}(\theta, \mathcal{M}) = 1$ then $\mathcal{M}$ has feature θ, otherwise $\mathcal{M}$ has not feature θ.

3.4 Modifying Theoretical Semantic System

Theoretical semantic system $\mathcal{U}^\mathcal{T} = \langle \underline{M},\ \mathcal{L}^\mathcal{T}_{\underline{M}},\ Cons \rangle$ is used instead of $\mathcal{S}^T = \langle Sent^T, \mathcal{U}^V_t, V^T, Val^T \rangle$ introduced in section 2. We assume that theoretical system $\mathcal{U}^\mathcal{T} = \langle \underline{M},\ \mathcal{L}^\mathcal{T}_{\underline{M}},\ Cons \rangle$ is related to observational semantic system $\mathcal{S}^\mathcal{T} = \langle \mathtt{M}^\mathcal{T},\ \mathcal{L}_\mathcal{C}, Val_\mathcal{C},\ \mathcal{L}_{\mathtt{M}^\mathcal{T}},\ Val_{\mathtt{M}^\mathcal{T}} \rangle$ introduced above. $\underline{M} = \bigcup \{Dom(\mathcal{M}) \mid \mathcal{M} \in \mathtt{M}^\mathcal{T}\}$ is a union of domains of all data matrices $\mathcal{M} \in \mathtt{M}^\mathcal{T}$. Language $\mathcal{L}^\mathcal{T}_{\underline{M}}$ is intended to express items of knowledge introduced in points (4) and (5) in section 3.1. There are lot of such items of knowledge [6, 8], several examples are in section 5.

We use function $Cons : (\mathcal{L}^\mathcal{T}_{\underline{M}} \times \mathtt{M}^\mathcal{T}) \rightarrow \mathcal{P}(\mathcal{L}_\mathcal{C})$. This function assigns to each couple $\langle \mathcal{I}, \mathcal{M} \rangle$ a set $Cons(\mathcal{I}, \mathcal{M})$ of formulas of language $\mathcal{L}_\mathcal{C}$. Here $\mathcal{I} \in \mathcal{L}^\mathcal{T}_{\underline{M}}$ is an item of knowledge (i.e. a formula of language $\mathcal{L}^\mathcal{T}_{\underline{M}}$) and $\mathcal{M} \in \mathtt{M}^\mathcal{T}$ is a data matrix of type $\mathcal{T}$ of related observational semantic system $\mathcal{S}^\mathcal{T}$. The set $Cons(\mathcal{I}, \mathcal{M})$ is considered as a set of all atomic consequences of item $\mathcal{I}$ of knowledge in data matrix $\mathcal{M}$, see point (C) in section 3.1.

4 Observational Semantic System of Association Rules

Observational semantic system $\mathcal{S}^\mathcal{T}_{\mathcal{AR}} = \langle \mathtt{M}^\mathcal{T}, \mathcal{L}^\mathcal{T}_{\mathcal{AR}}, Val^\mathcal{T}_{\mathcal{AR}}, \mathcal{L}_{\mathtt{M}^\mathcal{T}}, Val_{\mathtt{M}^\mathcal{T}} \rangle$ of type $\mathcal{T} = \langle t_1, \ldots, t_K \rangle$ concerning association rules is outlined in this section. Association rules of type $\mathcal{T}$ are couples of Boolean attributes created from columns of data matrices $\mathcal{M} \in \mathtt{M}^\mathcal{T}$. Language $\mathcal{L}^\mathcal{T}_{\mathcal{AR}}$ of association rules is introduced in section 4.1, evaluation function $Val^\mathcal{T}_{\mathcal{AR}}$ in section 4.2. Language $\mathcal{L}^\mathcal{T}_{\mathcal{AR}}$, set $\mathtt{M}^\mathcal{T}$ of data matrices and evaluation function $Val^\mathcal{T}_{\mathcal{AR}}$ constitute a logical calculus with important deduction rules, see section 4.3. All definitions are given informally.

We will not discuss here details of language $\mathcal{L}_{\mathtt{M}^\mathcal{T}}$ and related evaluation function $Val^\mathcal{T}_\mathtt{M}$. They are intended to express characteristics of particular data matrices. Examples: data matrix $\mathcal{M}_1$ concerns only pathological patients, data matrix $\mathcal{M}_2$ concerns patients from mountain region, etc. More detailed description is out of scope of this paper, additional research is assumed.

4.1 Language $\mathcal{L}^{\mathcal{T}}_{\mathcal{AR}}$ of Association Rules

An association rule is expression $\varphi \approx \psi$ where φ and ψ are Boolean attributes derived from columns of the analyzed data matrix and $\approx$ is a 4ft-quantifier [5]. Boolean attribute φ is called *antecedent* and ψ is called *succedent*. 4ft-quantifier defines relation of φ and ψ by associated function $F_{\approx}$ of $\approx$, see below.

Basic Boolean attributes are created first. The basic Boolean attribute is an expression $A(\alpha)$ where $\alpha \subset \{a_1, \ldots a_t\}$ and $\{a_1, \ldots a_t\}$ is the set of all categories of the attribute A. Here α is a *coefficient* of $A(\alpha)$. The basic Boolean attribute $A(\alpha)$ is true in row o of $\mathcal{M}$ if it is $A(o) \in \alpha$ where $A(o)$ is the value of the attribute A in row o. Boolean attributes φ and ψ are derived from basic Boolean attributes using connectives $\vee$, $\wedge$ and $\neg$ in the usual way. Examples of Boolean attributes are in figure 2.

$\mathcal{M}$	A_1	$\ldots$	A_K	$A_1(1)$	$A_K(2,6)$	$A_1(1) \wedge A_K(2,6)$
o_1	1	$\ldots$	6	1	1	1
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
o_n	1	$\ldots$	1	1	0	0

Data matrix and examples of Boolean attributes

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

4ft$(\varphi, \psi, \mathcal{M})$

Fig. 2. Derived Boolean attributes and 4ft-table 4ft$(\varphi, \psi, \mathcal{M})$

An example of association rule is expression $A_1(1) \wedge A_2(4,5) \approx A_K(2,6)$. Note that φ and ψ in $\varphi \approx \psi$ have no common attributes.

4.2 Evaluation Function $Val^{\mathcal{T}}_{\mathcal{AR}}$

Association rule $\varphi \approx \psi$ can be true or false in a given data matrix $\mathcal{M} \in \mathtt{M}^{\mathcal{T}}$. Rule $\varphi \approx \psi$ is verified on the basis of a *four-fold table* $4ft(\varphi,\psi, \mathcal{M})$ of φ and ψ in $\mathcal{M}$, see figure 2. Here a is the number of the objects (i.e. the rows of $\mathcal{M}$) satisfying both φ and ψ, b is the number of the objects satisfying φ and not satisfying ψ, and similarly for c and d, see figure 2. Four-fold table $4ft(\varphi,\psi, \mathcal{M})$ is written as $\langle a, b, c, d\rangle$ and called *4ft-table*.

Evaluation function $Val^{\mathcal{T}}_{\mathcal{AR}}$ assigns a value 0 or 1 to each couple $\langle \varphi \approx \psi, \mathcal{M}\rangle$ where $\varphi \approx \psi$ is the association rule and $\mathcal{M} \in \mathtt{M}^{\mathcal{T}}$. If $Val^{\mathcal{T}}_{\mathcal{AR}}(\varphi \approx \psi, \mathcal{M}) = 1$ then we say that *rule* $\varphi \approx \psi$ *is true in* $\mathcal{M}$ and if $Val^{\mathcal{T}}_{\mathcal{AR}}(\varphi \approx \psi, \mathcal{M}) = 0$ then we say that *rule* $\varphi \approx \psi$ *is false in* $\mathcal{M}$. $Val^{\mathcal{T}}_{\mathcal{AR}}(\varphi \approx \psi, \mathcal{M})$ is defined using 4ft-table 4ft$(\varphi, \psi, \mathcal{M})$ of φ and ψ in $\mathcal{M}$ and associated function $F_{\approx}$ of $\approx$.

Associated function $F_{\approx}$ *of 4ft quantifier* $\approx$ is a $\{0, 1\}$ - valued function defined for all quadruples $\langle a, b, c, d\rangle$ of natural numbers. Value $Val(\varphi \approx \psi, \mathcal{M})$ of association rule $\varphi \approx \psi$ in data matrix $\mathcal{M} \in \mathtt{M}^{\mathcal{T}}$ is defined such that $Val(\varphi \approx \psi, \mathcal{M}) = F_{\approx}(a, b, c, d)$ where $\langle a, b, c, d\rangle = 4ft(\varphi, \psi, \mathcal{M})$. Examples of 4ft-quantifiers and their associated functions are in table 1 where $0 < p \leq 1$ and $0 < \alpha < 0.5$ are real numbers, $B > 0$ is an integer number.

4ft-quantifier		Associated function $F_{\approx}(a,b,c,d)$
Name	Symbol $\approx$	$F_{\approx}(a,b,c,d) = 1$ iff
Founded implication	$\Rightarrow_{p,B}$	$\frac{a}{a+b} \geq p \wedge a \geq B$
Lower critical implication	$\Rightarrow^{!}_{p,\alpha,B}$	$\sum_{i=a}^{r} \binom{r}{i} (1-p)^{r-i} \leq \alpha \wedge a \geq B$
Founded equivalence	$\equiv_{p,B}$	$\frac{a+d}{a+b+c+d} \geq p \wedge a \geq B$
Fisher	$\approx_{\alpha,B}$	$\sum_{i=a}^{\min(r,k)} \frac{\binom{k}{i}\binom{n-k}{r-i}}{\binom{r}{n}} \leq \alpha \wedge a \geq B$
Above average dependence	$\sim^{+}_{q,B}$	$\frac{a}{a+b} \geq (1+q)\frac{a+c}{a+b+c+d} \wedge a \geq B$

Table 1. Examples of 4ft-quantifiers

4.3 Deduction Rules in Logical Calculus of Association Rules

Language $\mathcal{L}^{\mathcal{T}}_{\mathcal{AR}}$, set of data matrices $\mathtt{M}^{\mathcal{T}}$ and evaluation function $Val^{\mathcal{T}}_{\mathcal{AR}}$ constitute a logical calculus of association rules [5]. There are both theoretically interesting and practically useful results concerning logical calculi of association rules. Most of them are related to classes of 4ft-quantifiers [5]. An example of a class of 4ft-quantifiers is the class of implicational 4ft-quantifiers. It is defined such that 4ft-quantifier $\approx$ is implicational if it satisfies the condition: if $F_{\approx}(a,b,c,d) = 1 \wedge a' \geq a \wedge b' \leq b$ then also $F_{\approx}(a',b',c',d') = 1$. Both 4ft-quantifiers $\Rightarrow_{p,B}$ and $\Rightarrow^{!}_{p,\alpha,B}$ (see Table 1) are implicational.

Criteria of soundness of deduction rules $\frac{\varphi \approx \psi}{\varphi' \approx \psi'}$ where both $\varphi \approx \psi$ and $\varphi' \approx \psi'$ are association rules were found [5]. Criteria are related to important classes of association rules. We outline such criterion for the class of *interesting implicational quantifiers*. All practically important implicational 4ft-quantifiers are interesting implicational quantifiers.

If $\Rightarrow^{*}$ is an interesting implicational quantifier then there are formulas ω_{1A}, ω_{1B}, ω_2 of propositional calculus created from φ, ψ, φ', ψ' so that the deduction rule $\frac{\varphi \Rightarrow^{*} \psi}{\varphi' \Rightarrow^{*} \psi'}$ is sound if and only if at least one of the following conditions (1), (2) are satisfied: (1) – both ω_{1A} and ω_{1B} are tautologies, (2) – ω_2 is a tautology.

Similar theorems are proved for additional important classes of 4ft-quantifiers [5]. These results are crucial, see section 6.2.

5 Theoretical Semantic System of Association Rules

Theoretical semantic system $\mathcal{U}^{\mathcal{T}}_{\mathcal{AR}} = \langle \underline{M}, \mathcal{L}^{\mathcal{T}}_{\underline{M}}, Cons^{\mathcal{T}}_{\mathcal{AR}} \rangle$ related to observational semantic system $\mathcal{S}^{\mathcal{T}}_{\mathcal{AR}} = \langle \mathtt{M}^{\mathcal{T}}, \mathcal{L}^{\mathcal{T}}_{\mathcal{AR}}, Val^{\mathcal{T}}_{\mathcal{AR}}, \mathcal{L}_{\mathtt{M}^{\mathcal{T}}}, Val_{\mathtt{M}^{\mathcal{T}}} \rangle$ of type $\mathcal{T} = \langle t_1, \ldots, t_K \rangle$ concerning generalized association rules is sketched in this section.

Set $\underline{M}$ and language $\mathcal{L}^{\mathcal{T}}_{\underline{M}}$ are introduced in section 3.4. Language $\mathcal{L}^{\mathcal{T}}_{\underline{M}}$ is intended to express items of knowledge introduced in points (4) and (5) in section 3.1. We give four important examples of formulas of language $\mathcal{L}^{\mathcal{T}}_{\underline{M}}$. Then we outline how function $Cons^{\mathcal{T}}_{\mathcal{AR}}$ is constructed for one of these examples.

The examples are: $A \uparrow\uparrow B$, $A \uparrow\downarrow B$, $A \rightarrow^{+} \omega$, and $\omega_1 \rightarrow^{+} \omega_2$. Here A is one of attributes $A_1, \ldots, A_K$ of language $\mathcal{L}^{\mathcal{T}}_{\mathcal{AR}}$, the same is true for B. In addition,

ω, ω_1, ω_2 are Boolean attributes of $\mathcal{L}^{\mathcal{T}}_{\mathcal{AR}}$ and ω does not contain attribute A. Intuitive meaning of particular formulas:

- $A \uparrow\uparrow B$ means *if A increases then B increases too*
- $A \uparrow\downarrow B$ means *if A increases then B decreases*
- $A \rightarrow^{+} \omega$ means *if A increases then relative frequency of ω increases*
- $\omega_1 \rightarrow^{+} \omega_2$ means *if ω_1 is satisfied then relative frequency of ω_2 increases.*

We show how function $Cons^{\mathcal{T}}_{\mathcal{AR}}$ creates a set $Cons^{\mathcal{T}}_{\mathcal{AR}}(A \uparrow\uparrow B, \mathcal{M})$ of association rules – formulas of language $\mathcal{L}^{\mathcal{T}}_{\mathcal{AR}}$ which can be considered as a set of all atomic consequences of $A \uparrow\uparrow B$ in data matrix $\mathcal{M}$. Function $Cons^{\mathcal{T}}_{\mathcal{AR}}$ can be seen as a family of functions $Cons^{\mathcal{T}}_{\approx}$ where $\approx$ is a 4ft-quantifier of language $\mathcal{L}^{\mathcal{T}}_{\mathcal{AR}}$.

Function $Cons^{\mathcal{T}}_{\approx}$ creates a set $Cons^{\mathcal{T}}_{\approx}(A \uparrow\uparrow B, \mathcal{M})$ of association rules – formulas of language $\mathcal{L}^{\mathcal{T}}_{\mathcal{AR}}$ such that this set can be considered as a set of all atomic consequences of $A \uparrow\uparrow B$ of the form $\rho \approx \sigma$ in data matrix $\mathcal{M}$. Then it is $Cons^{\mathcal{T}}_{\mathcal{AR}}(A \uparrow\uparrow B, \mathcal{M}) = \bigcup\{Cons^{\mathcal{T}}_{\approx}(A \uparrow\uparrow B, \mathcal{M}) \mid \approx \text{ belongs to } \mathcal{L}^{\mathcal{T}}_{\mathcal{AR}}\}$.

We outline function $Cons^{\mathcal{T}}_{\Rightarrow_{p,B}}$ for 4ft-quantifier $\Rightarrow_{p,B}$ of founded implication (see Table 1) and item $A \uparrow\uparrow B$ of domain knowledge. Functions $Cons^{\mathcal{T}}_{\approx}$ for additional 4ft-quantifiers and formulas of $\mathcal{L}^{\mathcal{T}}_{M}$ are defined similarly [6].

We assume that attribute A has categories $1, \ldots, u$ and attribute B has categories $1, \ldots, v$. Our task is to define a set of rules $\rho \Rightarrow_{p,B} \sigma$ which can be naturally considered as a set of all the consequences of item $A \uparrow\uparrow B$ and which are as simple as possible. In this case, we consider the simplest rules to be in the form $A(\alpha) \Rightarrow_{p,B} B(\beta)$ where $\alpha \subset \{1, \ldots, u\}$ and $\beta \subset \{1, \ldots, v\}$.

Rule $A(low) \Rightarrow_{p,B} B(low)$ stating that "*if A is low then B is low*" can be understood as a natural consequence of $A \uparrow\uparrow B$. The only problem is to define the coefficients α and β that can be understood as "low". This can be done so that we choose natural A_{low}, $1 < A_{low} < u$ and natural B_{low}, $1 < B_{low} < v$ and then we consider α as "low" iff $\alpha \subset \{1, \ldots, A_{low}\}$ and β as "low" iff $\beta \subset \{1, \ldots, B_{low}\}$, see also section 6.1.

Also rule $A(high) \Rightarrow_{p,B} B(high)$ stating that "*if A is high then B is high*" can be understood as a natural consequence of $A \uparrow\uparrow B$. The coefficients α and β can be defined as "high" in the following way. We choose natural A_{high}, $1 < A_{low} < A_{high} < u$ and natural B_{high}, $1 < B_{low} < B_{high} < v$ and then we consider α as "high" iff $\alpha \subset \{A_{high}, \ldots, v\}$ and β as "high" iff $\beta \subset \{B_{high}, \ldots, v\}$.

It remains to define values of parameters p and B of $\Rightarrow_{p,B}$. A possibility is to allow each $p \geq 0.9$ and $B \geq \frac{n}{20}$ where n is the number of rows of data matrix $\mathcal{M}$. However, boundaries of p and B as well as values A_{low}, A_{high}, B_{low}, B_{high} should be determined by a domain expert. The set of rules $A(low) \Rightarrow_{p,B} B(low)$ and $A(high) \Rightarrow_{p,B} B(high)$ satisfying the above given conditions can be considered as $Cons^{\mathcal{T}}_{\Rightarrow_{p,B}}(A \uparrow\uparrow B, \mathcal{M})$ – a set of atomic consequences of $A \uparrow\uparrow B$ of the form $\rho \Rightarrow_{p,B} \sigma$ in $\mathcal{M}$.

Set $Cons^{\mathcal{T}}_{\Rightarrow_{p,B}}(A \uparrow\uparrow B, \mathcal{M})$ can be defined in a more precise way by adding rules $A(medium) \Rightarrow_{p,B} B(medium)$ with a suitable definition of "medium". Rules $A(low,\ medium) \Rightarrow_{p,B} B(medium)$, $A(low,\ medium) \Rightarrow_{p,B} B(medium,\ high)$, and $A(medium) \Rightarrow_{p,B} B(medium,\ high)$ can also be added.

Note: there is a natural requirement on the reasonable consistency of set $Cons^{\mathcal{T}}_{\mathcal{AR}}(A \uparrow\uparrow B, \mathcal{M})$ of atomic consequences of the $A \uparrow\uparrow B$ i.e. there cannot be two atomic consequences $\rho_1 \approx \sigma_1$ and $\rho_2 \approx \sigma_2$ that contradict each other. A detailed discussion of this topic is, however, without the scope of this paper.

6 4ft-Discoverer

We have defined observational system $\mathcal{S}^{\mathcal{T}}_{\mathcal{AR}} = \langle \mathtt{M}^{\mathcal{T}}, \mathcal{L}^{\mathcal{T}}_{\mathcal{AR}}, Val^{\mathcal{T}}_{\mathcal{AR}}, \mathcal{L}_{\mathtt{M}^{\mathcal{T}}}, Val_{\mathtt{M}^{\mathcal{T}}} \rangle$ and related theoretical system $\mathcal{U}^{\mathcal{T}}_{\mathcal{AR}} = \langle \underline{M}, \mathcal{L}^{\mathcal{T}}_{\underline{M}}, Cons^{\mathcal{T}}_{\mathcal{AR}} \rangle$. The goal of this section is to discuss possibilities of implementation of a theoretical framework for dealing with domain knowledge involved in these systems.

We use the GUHA procedure 4ft-Miner mining for association rules - couples of Boolean attributes created from columns of data matrices $\mathcal{M} \in \mathtt{M}^{\mathcal{T}}$ [7]. The 4ft-Miner procedure has very fine tools to define a set of association rules to be generated and verified. It deals, among other, with basic Boolean attributes $A(\alpha)$, $B(\beta)$, $A(low)$, $B(low)$ etc. Main features of 4ft-Miner procedure are outlined in section 6.1.

Intention to develop two additional analytical procedures G-FILTER and G-SYNT related to each GUHA procedure is announced in points (D) and (E) in section 3.1. We present principles of procedures 4ft-Filter and 4ft-Synt related to 4ft-Miner procedure. The 4ft-Filter procedure is intended to filter out consequences of a given item of domain knowledge from the output of 4ft-Miner. Item of domain knowledge is expressed by a formula of $\mathcal{L}_{\mathtt{M}^{\mathcal{T}}}$. The 4ft-Synt procedure is intended to recognize groups of patterns which can be considered as a consequence of a yet not known item of knowledge. Principles of both procedures are introduced in section 6.2.

Semantic systems $\mathcal{S}^{\mathcal{T}}_{\mathcal{AR}}$ and $\mathcal{U}^{\mathcal{T}}_{\mathcal{AR}}$ together with the procedures 4ft-Miner, 4ft-Filter, and 4ft-Synt constitute a framework for a process of data mining for association rules based on domain knowledge. We call this framework *4ft-Discoverer*, i.e. $4ft\mathcal{D}$. It is

$$4ft\mathcal{D} \;=\; \langle \mathcal{S}^{\mathcal{T}}_{\mathcal{AR}},\; \mathcal{U}^{\mathcal{T}}_{\mathcal{AR}},\; \text{4ft-Miner},\; \text{4ft-Filter},\; \text{4ft-Synt} \rangle\ .$$

6.1 4ft-Miner

The 4ft-Miner procedure mines for association rules $\varphi \approx \psi$, see section 4. Input parameters define 4ft-quantifier $\approx$, set of relevant antecedents Φ and set of relevant succedents Ψ; we assume $\varphi \in \Phi$ and $\psi \in \Psi$.

Each antecedent is a conjunction $\tau_1 \wedge \ldots \wedge \tau_m$ of *partial antecedents* $\tau_1, \ldots, \tau_m$. Each partial antecedent is either a conjunction $\lambda_1 \wedge \ldots \wedge \lambda_q$ or a disjunction $\lambda_1 \vee \ldots \vee \lambda_q$ of *literals* $\lambda_1, \ldots, \lambda_q$. Each literal is a basic Boolean attribute $A(\alpha)$ or its negation $\neg A(\alpha)$. Definition of a set of relevant antecedents Φ consists of definitions of relevant partial antecedents $\Phi_1, \ldots, \Phi_m$. Conjunction $\tau_1 \wedge \ldots \wedge \tau_m$ is a relevant antecedent if $\tau_1 \in \Phi_1, \ldots, \tau_m \in \Phi_m$.

Definition of a relevant partial antecedent is given by a list $A'_1, \ldots, A'_u$ of attributes, by a minimal and maximal number of literals in particular partial

antecedents and by a type of partial antecedent, i.e. *conjunctions* or *disjunctions*. In addition, for each attribute A' a set of relevant basic Boolean attributes which are automatically generated is defined. There are various detailed possibilities how to define all relevant basic Boolean attributes $A'(\alpha)$ [7]. We outline only one of them. We use attribute A with categories 1, 2, 3, 4, 5. Option *intervals of length 2-3* gives basic Boolean attributes $A(1,2)$, $A(2,3)$, $A(3,4)$, $A(4,5)$, $A(1,2,3)$, $A(2,3,4)$, $A(3,4,5)$. This way we can get basic Boolean attributes $A(low)$, $A(high)$, $B(low)$, $B(high)$, see section 5.

Set Ψ of relevant succedents is defined analogously. The 4ft-Miner procedure does not use well known a-priori algorithm [1]. Its implementation is based on representation of analyzed data by suitable strings of bits [7]. Its performance is good enough to solve a lot of practically important tasks. A detailed study of its time and space complexity is in [7].

6.2 4ft-Filter and 4ft-Synt

The 4ft-Filter procedure is intended to filter out consequences of a given item of domain knowledge from association rules produced by the 4ft-Miner procedure. Item of domain knowledge is represented by a formula $\mathcal{I} \in \mathcal{L}^{\mathcal{T}}_{\underline{M}}$, see section 3.4.

Function $Is4ftConsequence(\mathcal{I}, \varphi \approx \psi, \mathcal{M})$ defined for all formulas $\mathcal{I} \in \mathcal{L}^{\mathcal{T}}_{\underline{M}}$, association rules $\varphi \approx \psi \in \mathcal{L}^{\mathcal{T}}_{\mathcal{AR}}$ and data matrices $\mathcal{M} \in \mathtt{M}^{\mathcal{T}}$ can be used to realize the 4ft-Filter procedure. It is $Is4ftConsequence(\mathcal{I}, \varphi \approx \psi, \mathcal{M}) = 1$ if the rule $\varphi \approx \psi$ can be considered as a consequence of $\mathcal{I}$, otherwise it is $Is4ftConsequence(\mathcal{I}, \varphi \approx \psi, \mathcal{M}) = 0$.

Value $Is4ftConsequence(\mathcal{I}, \varphi \approx \psi, \mathcal{M})$ is computed using function $Cons^{\mathcal{T}}_{\mathcal{AR}}$, see section 5 and using deduction rules $\frac{\varphi \approx \psi}{\varphi' \approx \psi'}$, see section 4.3. There are criteria of correctness of rules $\frac{\varphi \approx \psi}{\varphi' \approx \psi'}$ for each 4ft-quantifier $\approx$ of 4ft-Miner procedure [5, 7]. Function $Cons^{\mathcal{T}}_{\mathcal{AR}}$ is defined for all $\mathcal{I} \in \mathcal{L}^{\mathcal{T}}_{\underline{M}}$, and $\mathcal{M} \in \mathtt{M}^{\mathcal{T}}$ such that $Cons^{\mathcal{T}}_{\mathcal{AR}}(\mathcal{I}, \mathcal{M}) = \Lambda$ and Λ is a set of all association rules $\rho \approx \sigma$ which can be considered as atomic consequences of $\mathcal{I}$ in $\mathcal{M}$.

Value $Is4ftConsequence(\mathcal{I}, \varphi \approx \psi, \mathcal{M})$ is computed in two steps. In the first step, we compute set $\Lambda = Cons^{\mathcal{T}}_{\mathcal{AR}}(\mathcal{I}, \mathcal{M})$. In the second step, we test correctness of $\frac{\rho \approx \sigma}{\varphi \approx \psi}$ for each $\rho \approx \sigma \in \Lambda$. If there is such a correct rule, then $\varphi \approx \psi$ is considered as a consequence of $\mathcal{I}$ in $\mathcal{M}$ and $Is4ftConsequence(\mathcal{I}, \varphi \approx \psi, \mathcal{M}) = 1$. Otherwise $Is4ftConsequence(\mathcal{I}, \varphi \approx \psi, \mathcal{M}) = 0$.

Function $Is4ftConsequence(\mathcal{I}, \varphi \approx \psi, \mathcal{M})$ can also be used to realize the procedure 4ft-Synt which recognizes group of rules $\varphi \approx \psi$, which can be considered as consequences of (yet unknown) items of knowledge. We assume that each even yet unknown item of knowledge is represented by a formula of language $\mathcal{L}^{\mathcal{T}}_{\underline{M}}$. The procedure 4ft-Synt can be then realized such that we choose formula $\omega \in \mathcal{L}^{\mathcal{T}}_{\underline{M}}$ and using function $Is4ftConsequence(\omega, \varphi \approx \psi, \mathcal{M})$ we pick up all consequences of ω from output of 4ft-Miner procedure. However, we have somehow to limit set of tested formulas $\omega \in \mathcal{L}^{\mathcal{T}}_{\underline{M}}$. A more detailed study of this problem is out of the scope of this paper.

7 Conclusions

The goal of this paper was to elaborate theoretically an approach for dealing with domain knowledge in mining association rules. It was done by modifications of logic of discovery developed in [2]. General requirements for such modifications were discussed in section 3.1.

Then a framework $4ftD = \langle \mathcal{S}^{\mathcal{T}}_{\mathcal{AR}}, \mathcal{U}^{\mathcal{T}}_{\mathcal{AR}}$, 4ft-Miner, 4ft-Filter, 4ft-Synt $\rangle$ for dealing with domain knowledge when mining association rules is described. Association rules are understood as interesting couples of Boolean attributes derived from columns of the analyzed data matrix. The Boolean attributes are derived from basic Boolean attributes by connectives $\wedge, \vee, \neg$, see section 4.1. The general form of the basic Boolean attributes is $A(\alpha)$. Here α is automatically generated subset of categories of A. It makes possible to deal with notions like $A(low)$ and $B(high)$. Implemented procedure 4ft-Miner produces such association rules [7].

The presented approach relates these rules to items of domain knowledge like $A \uparrow\uparrow B$ concerning non-Boolean attributes, see section 5. The procedures 4ft-Filter and 4ft-Synt are suggestedto deal with such items of domain knowledge when interpreting results of 4ft-Miner. They are being implemented.

No similar approach concerning association rules is known to the author. However, a comparison of the presented approach with ways of dealing with domain knowledge in additional data mining areas is still a challenge and a subject of further work. It requires both theoretical study and experiments with 4ft-Discoverer after its implementation.

References

1. Aggraval R. et al (1996) Fast Discovery of Association Rules. In: Fayyad U.M. et al (eds) Advances in Knowledge Discovery and Data Mining. AAAI Press, Menlo Park
2. Hájek P., Havránek T. (1978) Mechanizing Hypothesis Formation(Mathematical Foundations for a General Theory), Springer–Verlag 1978
3. Hájek P., Holeňa M., Rauch J. (2010) The GUHA method and its meaning for data mining. Journal of Computer and System Science, 76, pp. 34 – 48.
4. Yang Q., Wu X. (2006) 10 Challenging Problems in Data Mining Research, International Journal of Information Technology & Decision Making, Vol. 5, No. 4, 2006, pp. 597 – 604.
5. Rauch J. (2005) Logic of Association Rules. Applied Intelligence 22, pp. 9 – 28.
6. Rauch J. (2009) Considerations on Logical Calculi for Dealing with Knowledge in Data Mining. In Ras Z. W., Dardzinska A. (Eds.): Advances in Data Management. Springer, 2009, pp. 177 – 202
7. Rauch J., Šimůnek M (2005) An Alternative Approach to Mining Association Rules. In: Lin T. Y. et al. (eds) Data Mining: Foundations, Methods, and Applications, Springer-Verlag, pp. 219 – 238
8. Rauch J., Šimůnek M.(2009) Dealing with Background Knowledge in the SEWEBAR Project. In: Berendt B. et al.: *Knowledge Discovery Enhanced with Semantic and Social Information*. Berlin, Springer-Verlag, 2009, pp. 89 – 106