

Action Rules

Zbigniew W. Ras

University of North Carolina, Charlotte, USA

Angelina Tzacheva

University of North Carolina, Charlotte, USA

Li-Shiang Tsay

University of North Carolina, Charlotte, USA

INTRODUCTION

There are two aspects of interestingness of rules that have been studied in data mining literature, objective and subjective measures (Liu, 1997; Adomavicius & Tuzhilin, 1997; Silberschatz & Tuzhilin, 1995, 1996). Objective measures are data-driven and domain-independent. Generally, they evaluate the rules based on their quality and similarity between them. Subjective measures, including unexpectedness, novelty and actionability, are user-driven and domain-dependent.

A rule is actionable if user can do an action to his/her advantage based on this rule (Liu, 1997). This definition, in spite of its importance, is too vague and it leaves open door to a number of different interpretations of actionability. In order to narrow it down, a new class of rules (called action rules) constructed from certain pairs of association rules, has been proposed in Ras & Wieczorkowska (2000). A formal definition of an action rule was independently proposed in Geffner & Wainer (1998). These rules have been investigated further in Tsay & Ras (2004) and Tzacheva & Ras (2004). To give an example justifying the need of action rules, let us assume that a number of customers have closed their accounts at one of the banks. We construct, possibly the simplest, description of that group of people and next search for a new description, similar to the one we have, with a goal to identify a new group of customers from which no one left that bank. If these descriptions have a form of rules, then they can be seen as actionable rules. Now, by comparing these two descriptions, we may find the cause why these accounts have been closed and formulate an action, which if undertaken by the bank, may prevent other customers from closing their accounts. Such actions are stimulated by action rules and they are seen as precise hints for actionability of rules. For example, an action rule may say that by inviting people from a certain group of customers for a glass of wine by the bank, it is guaranteed that these customers will not close their accounts and they do not move to

another bank. Sending invitations by regular mail to all these customers or inviting them personally by giving them a call are examples of an action associated with that action rule.

In paper by Ras & Gupta (2002), authors assume that information system is distributed and its sites are autonomous. They show that it is wise to search for action rules at remote sites when action rules extracted at the client site cannot be implemented in practice (suggested actions are too expensive or too risky). The composition of two action rules, not necessary extracted at the same site, was defined in Ras & Gupta (2002). Authors gave assumptions guaranteeing the correctness of such a composition. One of these assumptions requires that semantics of attributes, including the interpretation of null values, have to be the same at both sites. This assumption is relaxed in Tzacheva & Ras (2004) since authors allow different granularities of the same attribute at involved sites. In the same paper, they introduce the notion of a cost and feasibility of an action rule. Usually, a number of action rules or chains of action rules can be applied to reclassify a certain set of objects. The cost associated with changes of values within one attribute is usually different than the cost associated with changes of values within another attribute. The strategy for replacing the initially extracted action rule by a composition of new action rules, dynamically built, was proposed in the paper by Tzacheva & Ras (2004). This composition of rules uniquely defines a new action rule and it was built with a goal to lower the cost of reclassifying objects supported by the initial action rule.

BACKGROUND

In the paper by Ras & Wieczorkowska (2000), the notion of an action rule was introduced. The main idea was to generate, from a database, special type of rules which basically form a hint to users showing a way to

reclassify objects with respect to some distinguished attribute (called a decision attribute). Clearly, each relational schema gives a list of attributes used to represent objects stored in a database. Values of some of these attributes, for a given object, can be changed and this change can be influenced and controlled by user. However, some of these changes (for instance “profit”) cannot be done directly to a decision attribute. In such a case, definitions of this decision attribute in terms of other attributes (called classification attributes) have to be learned. These new definitions are used to construct action rules showing what changes in values of some attributes, for a given class of objects, are needed to reclassify objects the way users want. But users may still be either unable or unwilling to proceed with actions leading to such changes. In all such cases, we may search for definitions of values of any classification attribute listed in an action rule. By replacing a value of such attribute by its definition extracted either locally or at remote sites (if system is distributed), we construct new action rules, which might be of more interest to business users than the initial rule.

MAIN THRUST

The technology dimension will be explored to clarify the meaning of actionable rules including action rules and extended action rules.

Action Rules Discovery in a Stand-alone Information System

An information system is used for representing knowledge. Its definition, given here, is due to Pawlak (1991).

By an information system we mean a pair $S = (U, A)$, where:

1. U is a nonempty, finite set of objects (object identifiers),
2. A is a nonempty, finite set of attributes, that is, $a:U \rightarrow V_a$ for $a \in A$, where V_a is called the domain of a .

Information systems can be seen as decision tables. In any decision table together with the set of attributes a partition of that set into conditions and decisions is given. Additionally, we assume that the set of conditions is partitioned into stable and flexible conditions (Ras & Wiczorkowska, 2000).

Attribute $a \in A$ is called stable for the set U if its values assigned to objects from U can not change in time.

Otherwise, it is called flexible. “Date of Birth” is an example of a stable attribute. “Interest rate” on any customer account is an example of a flexible attribute. For simplicity reasons, we will consider decision tables with only one decision. We adopt the following definition of a decision table:

By a decision table we mean an information system $S = (U, A_1 \cup A_2 \cup \{d\})$, where $d \notin A_1 \cup A_2$ is a distinguished attribute called decision. The elements of A_1 are called stable conditions, whereas the elements of $A_2 \cup \{d\}$ are called flexible conditions. Our goal is to change values of attributes in A_1 for some objects from U so the values of the attribute d for these objects may change as well. Certain relationships between attributes from A_1 and the attribute d will have to be discovered first.

By $\text{Dom}(r)$ we mean all attributes listed in the IF part of a rule r extracted from S . For example, if $r = [(a1,3)*(a2,4) \rightarrow (d,3)]$ is a rule, then $\text{Dom}(r) = \{a1, a2\}$. By $d(r)$ we denote the decision value of rule r . In our example $d(r) = 3$.

If $r1, r2$ are rules and $B \subseteq A_1 \cup A_2$ is a set of attributes, then $r1/B = r2/B$ means that the conditional parts of rules $r1, r2$ restricted to attributes B are the same.

For example, if $r1 = [(a1,3) \rightarrow (d,3)]$, then $r1/\{a1\} = r/\{a1\}$.

Assume also that $(a, v \rightarrow w)$ denotes the fact that the value of attribute a has been changed from v to w . Similarly, the term $(a, v \rightarrow w)(x)$ means that $a(x)=v$ has been changed to $a(x)=w$. Saying another words, the property (a,v) of an object x has been changed to property (a,w) . Assume now that rules $r1, r2$ have been extracted from S and $r1/A_1 = r2/A_1$, $d(r1)=k1$, $d(r2)=k2$ and $k1 < k2$. Also, assume that $(b1, b2, \dots, bp)$ is a list of all attributes in $\text{Dom}(r1) \cap \text{Dom}(r2) \cap A_2$ on which $r1, r2$ differ and $r1(b1)=v1, r1(b2)=v2, \dots, r1(bp)=vp, r2(b1)=w1, r2(b2)=w2, \dots, r2(bp)=wp$.

By $(r1, r2)$ -action rule on $x \in U$ we mean a statement:

$$[(b1, v1 \rightarrow w1) \wedge (b2, v2 \rightarrow w2) \wedge \dots \wedge (bp, vp \rightarrow wp)](x) \Rightarrow [(d, k1 \rightarrow k2)](x).$$

If the value of the rule on x is true then the rule is valid. Otherwise it is false.

Let us denote by $U^{<r1>}$ the set of all customers in U supporting the rule $r1$. If $(r1, r2)$ -action rule is valid on $x \in U^{<r1>}$ then we say that the action rule supports the new profit ranking $k2$ for x .

To define an extended action rule (Ras & Tsay, 2003), let us assume that two rules are considered. We present

Table 1.

A (St)	B (Fl)	C (St)	E (Fl)	G (St)	H (Fl)	D (Decision)
a1	b1	c1	e1			d1
a1	b2			g2	h2	d2

3 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/action-rules/10555

Related Content

XML Object Identification

(2014). *Innovative Techniques and Applications of Entity Resolution* (pp. 140-170).

www.irma-international.org/chapter/xml-object-identification/103247

Mining Group Differences

Shane M. Butler and Geoffrey I. Webb (2005). *Encyclopedia of Data Warehousing and Mining* (pp. 795-799).

www.irma-international.org/chapter/mining-group-differences/10705

Path Mining in Web Processes Using Profiles

Jorge Cardoso (2005). *Encyclopedia of Data Warehousing and Mining* (pp. 896-901).

www.irma-international.org/chapter/path-mining-web-processes-using/10723

A Single Pass Algorithm for Discovering Significant Intervals in Time-Series Data

Sagar Savla and Sharma Chakravarthy (2008). *Data Warehousing and Mining: Concepts, Methodologies, Tools, and Applications* (pp. 3272-3284).

www.irma-international.org/chapter/single-pass-algorithm-discovering-significant/7833

Enhancing Web Search through Web Structure Mining

Ji-Rong Wen (2005). *Encyclopedia of Data Warehousing and Mining* (pp. 443-447).

www.irma-international.org/chapter/enhancing-web-search-through-web/10638