Constraint-Based Association Mining?

A data mining procedure can uncover thousands of rules from a given set of information, most of which end up being independent or tedious to the users. Users have a best sense of which "direction" of mining can lead to interesting patterns and the "form" of the patterns or rules they can like to discover.

Therefore, a good heuristic is to have the users define such intuition or expectations as constraints to constrain the search space. This strategy is called constraint-based mining.

Constraint-based algorithms need constraints to decrease the search area in the frequent itemset generation step (the association rule generating step is exact to that of exhaustive algorithms).

The general constraint is the support minimum threshold. If a constraint is uncontrolled, its inclusion in the mining phase can support a significant reduction of the exploration space because of the definition of a boundary inside the search space, following which exploration is not needed.

The importance is well defined − they create only association rules that are appealing to users.

The constraints can include the following which are as follows −

Knowledge type constraints − These define the type of knowledge to be mined, including association or correlation.

Data constraints − These define the set of task-relevant information such as Dimension/level constraints − These defines the desired dimensions (or attributes) of the information, or methods of the concept hierarchies, to be utilized in mining.

Dimension/level constraints: These specify the desired dimensions (or attributes) of the data, the abstraction levels, or the level of the concept hierarchies to be used in mining.

Interestingness constraints − These defines thresholds on numerical measures of rule interestingness, including support, confide ccscsnce, and correlation.

Rule constraints – These define the form of rules to be mined. Such constraints can be defined as metarules (rule templates), as the maximum or minimum number of predicates that can appear in the rule antecedent or consequent, or as relationships between attributes, attribute values, and/or aggregates.

The following constraints can be described using a high-level declarative data mining query language and user interface. This form of constraint-based mining enables users to define the rules that they can like to uncover, thus by creating the data mining process more efficient.

Furthermore, a sophisticated mining query optimizer can be used to deed the constraints defined by the user, thereby creating the mining process more effective. Constraint-based mining boost interactive exploratory mining and analysis.

Metarule-Guided Mining of Association Rules

"How are metarules useful?" Metarules allow users to specify the syntactic form of rules that they are interested in mining. The rule forms can be used as constraints to help improve the efficiency of the mining process. Metarules may be based on the analysist's experience, expectations, or intuition regarding the data or may be automatically generated based on the database schema.

Suppose that as a market analyst for AllElectronics you have access to the data describing customers (e.g., customer age, address, and credit rating) as well as the list of customer transactions. You are interested in finding associations between customer traits and the items that customers buy. However, rather than finding all of the association rules reflecting these relationships, you are interested only in determining which pairs of customer traits promote the sale of office software. A metarule can be used to specify this information describing the form of rules you are interested in finding. An example of such a metarule is

$$P_1(X, Y) \wedge P_2(X, W) \Rightarrow buys(X, \text{"office software"}), \qquad (7.11)$$

where P1 and P2 are predicate variables that are instantiated to attributes from the given database during the mining process, X is a variable representing a customer, and Y and W take on values of the attributes assigned to P1 and P2, respectively. Typically, a user will specify a list of attributes to be considered for instantiation with P1 and P2. Otherwise, a default set may be used. In general, a metarule forms a hypothesis regarding the relationships that the user is interested in probing or confirming. The data mining system can then search for rules that match the given metarule. For instance, Rule (7.12) matches or complies with Metarule.

$$age(X, \text{“30..39”}) \wedge income(X, \text{“41}K\text{..60}K\text{”}) \Rightarrow buys(X, \text{“office software”}). \qquad (7.12)$$

“How can metarules be used to guide the mining process?” Let’s examine this problem closely. Suppose that we wish to mine interdimensional association rules. A metarule is a rule template of the form

$$P_1 \wedge P_2 \wedge \cdots \wedge P_l \Rightarrow Q_1 \wedge Q_2 \wedge \cdots \wedge Q_r, \qquad (7.13)$$

where Pi (i = 1,..., l) and Qj (j = 1,..., r) are either instantiated predicates or predicate variables. Let the number of predicates in the metarule be p = l + r. To find interdimensional association rules satisfying the template, We need to find all frequent p-predicate sets, Lp. We must also have the support or count of the l-predicate subsets of Lp to compute the confidence of rules derived from Lp. This is a typical case of mining multidimensional association rules. By extending such methods using the constraint-pushing techniques described in the following section, we can derive efficient methods for metarule-guided mining.

References:

1. https://www.youtube.com/watch?v=wmzpgKeI8QI
2. https://tinyurl.com/yh8zan6m