Meta-Rule-Guided Mining of Association Rules in Relational Databases [*]

Yongjian Fu   Jiawei Han

School of Computing Science, Simon Fraser University, Canada

E-mail: {yongjian, han}@cs.sfu.ca

Abstract

A meta-rule-guided data mining approach is proposed and studied which applies meta-rules as a guidance at finding multiple-level association rules in large relational databases. A meta-rule is a rule template in the form of "$P_1 \wedge \cdots \wedge P_m \rightarrow Q_1 \wedge \cdots \wedge Q_n$", in which some of the predicates (and/or their variables) in the antecedent and/or consequent of the meta-rule could be instantiated. The rule template is used to describe what forms of rules are expected to be found from the database, and such a rule template is used as a guidance or constraint in the data mining process. Note that the predicate variables in a meta-rule can be instantiated against a database schema, whereas the variables or some high-level constants inside a predicate can be bound to multiple (but more specific) levels of concepts in the corresponding conceptual hierarchies. The concrete rules at different concept levels are discovered by a progressive deepening data mining technique similar to that developed in our study of mining multiple-level association rules. Two algorithms are developed along this line and a performance study is conducted to compare their relative efficiencies. Our experimental and performance studies demonstrate that the method is powerful and efficient in data mining from large databases.

1   Introduction

Database mining, i.e., the discovery of interesting knowledge from large amounts of data stored in databases, is a highly demanding and promising research topic because of its strong application potential and the wide availability of the huge amounts of data in databases.

Many methods have been proposed and developed in recent studies on data mining by development and integration of database, machine learning, and statistics techniques [9, 3]. However, a frequently encountered phenomenon in database mining is that although a mining system may discover a quite large number of rules, many of them could be poorly focused or lack of interest to users. Two major factors may contribute to this phenomenon: (1) lack of focus on the set of data to be studied, and (2) lack of constraints on the forms and/or kinds of rules or knowledge to be discovered.

The first problem, *the lack of focus on the set of data to be studied*, can be handled by introducing a data mining interface which specifies the set of data relevant to a particular mining task. For example, the DBMiner (previously DBLearn) system developed in our previous study [7] uses an SQL-like interface to specify the task-relevant set of data for a data mining query. Thus, in order to find the general characteristics of computer science graduate students in Canada, a where-clause is used to retrieve only those students of interest.

However, the second problem, *the lack of constraints on the forms and/or kinds of rules or knowledge to be discovered*, is not so straightforward to solve. There are many ways to specify the kinds of knowledge or the forms of rules to be discovered. For example, one may specify the types of knowledge to be discovered, such as characteristic rules, classification rules, association rules, and so on [7], or specify the number of disjuncts in a generalized rule, i.e., the expected (or maximum) number of distinct values of each generalized attribute or the number of tuples in the generalized relation [4]. Moreover, one may also specify some syntactic or semantic constraints on the forms of discovered rules[2, 8].

Recently, Shen et al. [10] proposed an interesting technique to specify the form of rules to be discovered in data mining, called *metaquery*, which presents a desired logical form for the rules to be discovered and serves as an important interface between human discoverers and the discovery system. In their initial study of metaquery-directed data mining [10], the rules to be discovered are confined to single concept level, whereas the knowledge discovery method is confined to Bayesian Data Cluster linked with a deductive database system $\mathcal{LDL}++$. Based on our observation, the scope of metaquery-directed mining could be substantially extended if the discovery of rules at multiple concept levels is explored [6]. Moreover, since a metaquery and its instantiated rules are in the form of association rules, the performance could be substantially enhanced if the database-oriented association rule mining algorithms [2] are adopted in the data mining process.

In this study, issues for meta-rule guided mining of multiple-level association rules are studied and a set of efficient mining algorithms are developed and tested. The study shows that the integration of meta-rule guided knowledge mining with the mining of multiple-level association rules enhances both the power and performance of a data mining system and thus is an interesting direction to pursue.

The remaining of the paper is organized as follows. In Section 2, preliminary concepts about meta-rule guided mining of multiple-level association rules are introduced, starting with some motivating examples. In Section 3, methods for mining meta-rule-guided single-variable rules are studied. In Section 4, methods for mining meta-rule-guided multiple-variable rules are examined. Variation of methods and other relevant issues on meta-rule-guided data mining are discussed in Section 5, and the study is concluded in Section 6.

2 Preliminary Concepts

To simplify our discussion, a relational model is adopted in our study, however, the methods developed here can be applied with some modifications to other data models, including extended-relational and object-oriented ones.

For effective data mining, a particular user is usually interested in only a subset of the data stored in a large database. An SQL-like data mining query[7] submitted to a data mining system should be first transformed into two portions: a *data collection* portion and a *knowledge discovery* portion. The former is essentially an SQL-query which will be executed against the database to collect the interested set of data. The latter, i.e., the knowledge discovery portion, will be examined in detail.

Example 2.1 Suppose that a portion of the relational schema of a university database is presented as follows.

```
student(name, sno, status, major, gpa, birth_date,
          birth_place, address)
course(cno, title, dept)
grading(sno, cno, instructor, semester, grade)
```

Let a data mining query (q_1) be presented as follows, which is to find the relationships between the attributes *status*, *gpa*, *birth_place*, and *address*, in relevance to *major*, for the students born in Canada.

(q_1) : discover rules in the form of
$$major(s : student, x) \land Q(s, y) \rightarrow R(s, z)$$
from student
where birth_place = "Canada"
in relevance to major, gpa, status, birth_place, address

The meta-rule of (q_1), "$major(s : student, x) \land Q(s, y) \rightarrow R(s, z)$", specifies the form of the rules to be

discovered, that is, each rule to be discovered is a logic rule containing two binary predicates, $major(s, x)$ and $Q(s, y)$, serving as the antecedent and one binary predicate, $R(s, z)$, as the consequent, with all the predicates sharing the first variable s which is the key of the relation *student*. Q and R are two predicate variables which can be instantiated by a list of relevant attributes: *gpa*, *status*, *birth_place*, and *address*.

By data mining techniques, the following rules may be discovered from the database.

$$major(s, \text{``Science''}) \land gpa(s, \text{``Excellent''}) \rightarrow$$
$$status(s, \text{``Graduate''}) \quad (60\%) \quad (2.1)$$
$$major(s, *) \land birth\_place(s, \text{``B.C.''}) \rightarrow$$
$$address(s, \text{``Burnaby''}) \quad (55\%) \quad (2.2)$$

Rule (2.1) indicates that 60% of the students majoring in science and having excellent gpa are graduate students and rule (2.2) indicates that 55% of the students majoring in anything and born in B.C. are living in the city of Burnaby.

The rules expressed by even lower level concepts, such as rules (2.3) to (2.4), can be further discovered if multiple-level information can be mined from the database. The semantic meaning of these rules is self-explanatory.

$$major(s, \text{``Physics''}) \land gpa(s, \text{``3.8\_4.0''}) \rightarrow$$
$$status(s, \text{``M.Sc''}) \quad (76\%) \quad (2.3)$$
$$major(s, \text{``CS''}) \land birth\_place(s, \text{``B.C.''}) \rightarrow$$
$$address(s, \text{``Burnaby''}) \quad (85\%) \quad (2.4)$$

Moreover, the associations among several relations can be discovered by joining these relations together. The relational joins can be explicitly expressed in the meta-rules as presented in the following data mining query (q_2).

(q_2) : discover rules in the form of
$$major(s, x) \land P(c, y) \rightarrow Q(s : S, c : C, z).$$
from student S, grading G, course C
where S.birth_place = "Foreign"

The query is to find the relationships among three predicates, one of which is instantiated to $major(s, x)$, the second contains the key of the *course* relation, and the third one, the consequent predicate, contains two key components from two relations: *student* and *course*, for the relevant set of the data: the students born in foreign countries.

By mining rules from multiple concept levels, the following rules may be discovered from the database.

$$major(s, \text{``Science''}) \land dept(c, \text{``CS''}) \rightarrow$$
$$grade(s, c, \text{``Good''}) \quad (60\%) \quad (2.5)$$

$$major(s, \text{``}Math\text{''}) \land cno(c, \text{``}CS\_400\_level\text{''}) \rightarrow$$
$$grade(s, c, \text{``}A-\text{''}) \quad (42\%) \qquad (2.6)$$

In Example 2.1, both the data mining queries and the discovered rules contain concepts at nonprimitive levels, i.e., levels higher than those stored in databases, such as "Science", "Graduate", "Excellent", etc. The high level concepts appearing in the query help the collection of the relevant set of data, whereas the concepts organized at different levels help progressively deeping the data mining process by first browsing the high-level data and then mining detailed regularities at low levels.

In this study, we assume that multiple levels of concepts are organized in the form of concept hierarchies which are provided in the system for mining rules at multiple concept levels, however, the concept hierarchies can also be dynamically adjusted and/or automatically generated for flexible data mining [5].

To confine our study, we assume the rules to be discovered are conjunctive rules, i.e., a set of conjuncts in both the rule head and body. Moreover, the predicate variable in the meta-rules can only be instantiated against database schema (attributes). Furthermore, each predicate variable in a meta-rule is different from others and is instantiated to a distinct and different predicate name. Some relaxations of these restrictions will be discussed in Section 5.

As a notational convention, a predicate name starting with an upper-case letter represents a predicate variable. It can be instantiated by binding it to a concrete attribute name (which starts with a lower-case letter) in the schema. For example, a predicate variable $P(x, y)$ can be instantiated to $status(x, \text{``}Graduate\text{''})$ in Example 2.1.

Definition 2.1 A meta-rule is a rule template in the form of

$$P_1 \land P_2 \cdots \land P_m \rightarrow Q_1 \land Q_2 \land \cdots \land Q_n. \qquad (2.7)$$

where P_i (for $i = 1, \ldots, m$) and Q_j (for $j = 1, \ldots, n$) are either instantiated predicates or predicate variables.

The rule "$major(s, x) \land P(c, y) \rightarrow Q(s : S, c : C, z)$" in Example 2.1 is a meta-rule.

Definition 2.2 A rule, R_c, complies with a meta-rule, R_M, if and only if it can be unified with R_M.

For example, rule (2.5) complies with the meta-rule "$major(s, x) \land P(c, y) \rightarrow Q(s : S, c : C, z)$" in Example 2.1.

Definition 2.3 A pattern, p, is one predicate p_i or a set of conjunctive predicate $p_i \land \cdots \land p_j$, where p_i, $\ldots$, p_j are predicates instantiated against the database schema.

The support of a pattern p in a set S, $\sigma(p/S)$, is the number of the tuples in S which contain p versus the total number of tuples in S. The confidence of $p \rightarrow q$ in S, $\varphi(p \rightarrow q/S)$, is the ratio of $\sigma(p \land q/S)$ versus $\sigma(p/S)$, i.e., the probability that pattern q also occurs in S when pattern p occurs in S.

To find relatively frequently occurring patterns and reasonably strong rule implications, a user or an expert may specify two thresholds: *minimum support*, σ', and *minimum confidence*, φ'. Notice that for finding multiple-level association rules, different minimum support and/or minimum confidence can be specified at different levels.

Definition 2.4 A pattern p is large in set S at level l if the support of p is no less than its corresponding minimum support threshold σ'_l. The confidence of a rule "$p \rightarrow q/S$" is high at level l if its confidence is no less than its corresponding minimum confidence threshold φ'_l.

Definition 2.5 A rule "$p \rightarrow q/S$" is strong if, for a set S, each ancestor (i.e., the corresponding high level predicate) of every predicate in p and q, if any, is large at its corresponding level, "$p \land q/S$" is large (at the current level), and the confidence of "$p \rightarrow q/S$" is high (at the current level).

The definition indicates that if "$p \rightarrow q/S$" is strong, then (1) $\sigma(p \land q/S) \geq \sigma'$, (and thus, $\sigma(p/S) \geq \sigma'$, and $\sigma(q/S) \geq \sigma'$), and (2) $\varphi(p \rightarrow q/S) \geq \varphi'$, at its corresponding level. It also represents a filtering process which confines the patterns to be examined at lower levels to be only those with large supports at their corresponding high levels (and thus avoids the generation of many meaningless combinations formed by the descendants of the small patterns). For example, in a data set related to $major(s, y)$, if "$(y =)$ science" is a large pattern, its lower level patterns such as "physics" will be examined; whereas if "art" is small, its descendants such as "performance art" will not be examined further.

Based on the two mining queries presented in Example 2.1, meta-rule guided mining of multiple-level association rules can be classified into two categories: (1) mining single-variable association rules, and (2) mining multiple-variable association rules. The former discovers association rules in the form like (2.3), in which each predicate contains only one and the same variable; whereas the latter discovers rules in the form like (2.5), in which some predicate(s) may contain more than one variable, which may often involve join(s) of more than one relation.

3 Meta-rule-guided mining of single-variable rules

In this section, we examine the methods for meta-rule guided mining of single-variable association rules. A single-variable association rule represents an association relationship among a set of properties in a data relation at different concept levels.

Definition 3.1 A single-variable meta-rule is in the form of

$$P_1(t : rel, x_1) \wedge \cdots \wedge P_m(t, x_m) \rightarrow$$
$$Q_1(t, y_1) \wedge \cdots \wedge Q_n(t, x_n) \qquad (3.1)$$

where P_i (for $i = 1, \ldots, n$) and Q_j (for $j = 1, \ldots, m$) are either instantiated predicates or predicate variables, and the common variable t represents the key of a relation rel. □

By data mining, each predicate variable in a discovered rule will be instantiated to a concrete predicate name which is an attribute name of the relation rel, the common variable t will remain to be a variable which is an abstraction of the key or key component of the relation, and other variables in the predicates will be instantiated to the high-level or primitive level constants (i.e., properties) of the corresponding predicates (attributes).

For example, the meta-rule "$major(s : student, x) \wedge Q(s, y) \rightarrow R(s, z)$" in ($q_1$) of Example 2.1 is a single-variable meta-rule, and the discovered rule (2.1) indicates that the common variable s remains to be a variable which is an abstraction of the key of the relation $student$, and other variables in the predicates are instantiated to constants, such as *Science*, *Excellent*, and *Graduate* in the corresponding predicates, such as *major*, *gpa*, and *status*, respectively.

For efficient mining of multiple-level single-variable association rules, two techniques: a *large-predicate growing technique* and a *p-predicate testing technique*, are proposed and examined in the next two subsections.

3.1 A large-predicate growing technique

Following our previous study on mining multiple-level association rules [6], a large-predicate growing technique is proposed as follows.

First, the set of relevant data is collected into an initial data relation by executing an SQL query specified by the data mining query. Second, large 1-predicate-sets, $\mathcal{L}[1, 1]$, $\mathcal{L}[2, 1]$, ..., $\mathcal{L}[max\_l, 1]$, are derived at each concept level (from the top-most desired concept level, level 1 down to level max_l) by scanning the initial data relation once, where level max_l is the lowest level where a non-empty large 1-predicate-set can be derived. Third, large 2-predicate-sets are derived at each concept level by first generating the candidate large 2-predicate-sets and then scanning the initial data relation to compute the large 2-predicate-sets. Fourth, this process continues until the large p-predicate-sets are derived at each concept level, where p is the total number of predicates in the meta-rule, i.e., $p = m + n$ in rule (3.1). Finally, the rules in the form of meta-rules are generated from the large p-predicate-sets at each concept level based on the specified confidence threshold at this level.

This technique is illustrated in the following example.

Example 3.1 We examine how to derive the multiple-level strong association rules for query (q_1) of Example 2.1.

1. The initial data relation $\mathcal{R}_0$ (a fragment shown in Table 1) is derived by performing selection to collect the students who were born in Canada and then projection on the set of relevant attributes: *major, gpa, status, birth_place*, and *address*.

major	gpa	status	birth_place	address
CS	3.85	Senior	Vancouver, B.C., Canada	123 Curtis, Burnaby, B.C., Canada
...	...	...	...	...

Table 1: A fragment of *student* relation in relevance to the data mining task

2. Large 1-predicate-set tables at multiple concept levels, (as shown in Table 2), i.e., $\mathcal{L}[1, 1]$, $\mathcal{L}[2, 1]$, ..., $\mathcal{L}[max\_l, 1]$, are derived by scanning the initial data relation $\mathcal{R}_0$ once.

$\mathcal{L}[1, 1]$		$\mathcal{L}[2, 1]$		$\mathcal{L}[3, 1]$	
major	count	major	count	major	count
Science	4,850	Appl._Sci.	1,364	CS	675
...	...	...	...	...	...
gpa	count	gpa	count	gpa	count
Excellent	2,173	3.8_4.0	1,731	3.8_3.9	1,043
...	...	...	...	...	...
status	count	status	count	status	count
Underg.	20,204	Senior	4,204	Senior	4,204
...	...	...	...	...	...

Table 2: A fragment of large 1-predicate tables at different concept levels

3. Large 2-predicate-sets at multiple concept levels (as shown in Table 3), i.e., $\mathcal{L}[1, 2]$, $\mathcal{L}[2, 2]$, ..., $\mathcal{L}[max\_l, 2]$, are derived by first generating the candidate large 2-predicate-sets and then scanning $\mathcal{R}_0$ to compute the large 2-predicate-sets.

$\mathcal{L}[1,2]$

major	gpa	count
Science	Excellent	819
...	...	...
major	status	count
Science	Underg.	6,914
...	...	...

$\mathcal{L}[2,2]$

major	gpa	count
Appl._Sci.	3.8_4.0	327
...	...	...
major	status	count
Appl._Sci.	Senior	2,149
...	...	...

$\mathcal{L}[3,2]$

major	gpa	count
CS	3.8_3.9	174
...	...	...
major	status	count
CS	Senior	891
...	...	...

Table 3: A fragment of large 2-predicate tables at different concept levels

4. This process continues until the large p-predicate-sets at multiple concept levels, i.e., $\mathcal{L}[1,p]$, $\mathcal{L}[2,p]$, ..., $\mathcal{L}[max\_l,p]$, where p is the total number of predicates in the meta-rule, are derived. The tables so derived for the large 3-predicate sets are presented in Table 4.

$\mathcal{L}[1,3]$

major	gpa	status	count
Science	Excellent	Underg.	526
...	...	...	...

$\mathcal{L}[2,3]$

major	gpa	status	count
Appl._Sci.	3.8_4.0	Senior	274
...	...	...	...

$\mathcal{L}[3,3]$

major	gpa	status	count
CS	3.8_3.9	Senior	180
...	...	...	...

Table 4: A fragment of large 3-predicate tables at different concept levels

Rule	Support	Confidence
...	...	...
$major(s, \text{``Science''}) \wedge$ $birth\_place(s, \text{``Western\_Canada''})$ $\rightarrow address(s, \text{``B.C.''})$	25%	95%
...	...	...
$major(s, \text{``CS''}) \wedge$ $gpa(s, \text{``3.8\_3.9''})$ $\rightarrow status(s, \text{``Senior''})$	5%	25.6%
...	...	...

Table 5: Rules generated from the large 3-predicate tables at different concept levels

5. The rules in the form of meta-rules are generated in Table 5 from the large 3-predicate-sets at multiple concept levels, based on the specified confidence threshold at each level. □

The above example leads to the following algorithm for mining meta-rule guided single-variable strong ML-association rules using large predicate growing technique.

Algorithm 3.1 (large predicate-growing)
Meta-rule guided mining of single-variable strong ML-association rules using large predicate growing technique.

Input: (1) $\mathcal{DB}$, a relational database, (2) $\mathcal{H}$, a concept hierarchy, (3) $minsup[l]$, the minimum support threshold, and $minconf[l]$, the minimum confidence threshold, for each concept level l, and (4) $meta\_R$, the meta-rule in the form of (3.1).

Output: Multiple-level strong association rules in the form of (3.1) discovered in relational database $\mathcal{DB}$.

Method: A top-down, progressively deepening process which collects large predicate sets at different concept levels as follows.

1. The initial data relation $\mathcal{R}_0$ is derived by executing an SQL query specified by the data mining query.

2. Large 1-predicate-set tables at each concept level, i.e., $\mathcal{L}[1,1]$, $\mathcal{L}[2,1]$, ..., $\mathcal{L}[max\_l,1]$, are derived by scanning the initial data relation $\mathcal{R}_0$ once. Note that a predicate $p_i(t, c_i)$ is large at level l (and thus being included in $\mathcal{L}[l,1]$) if (1) its support is low less than than $minsup[l]$, and its corresponding concept c_i' at a higher-level $l-1$ is large.

3. Derive the large k-predicate-set tables at each concept level and for each k from 2 to p, i.e., derive $\mathcal{L}[l,k]$, for $l=1,\ldots,max\_l$, and $k=2,\ldots,p$, where p is the total number of predicates in the meta-rule.

   Note that a set of k predicates is large at level l if (1) each of its k subsets of $(k-1)$ predicates is large at level l, and (2) the support of the k predicates at level l is no less than $minsup[l]$.

4. For each concept level l, generate the rules in the form of meta-rules from the large p-predicate set tables $\mathcal{L}[l,p]$ if the confidence of the rule is no less than $minconf[l]$, the specified confidence threshold at this level. □

3.2 A direct p-predicate testing technique

The previous algorithm is a natural extension of the method developed in the study of mining multiple-level association rules [6]. A major difference of the requirements in meta-rule guided mining from that in the mining of general multiple-level association rules is that p, the number of large predicates in the rules to be generated, is predefined by the given meta-rule. This heuristic

can be used in the development of the variations of the rule mining algorithms.

Here we consider one variation of the mining technique: a *direct p-predicate generation and testing* technique. At the third step of Algorithm 3.1, instead of deriving large 2-predicate-sets at each concept level, and then large 3-predicates, etc., p-predicate sets are generated directly from the large 1-predicate sets and tested against the support threshold at each level. This technique is illustrated in the following similar example, followed by the algorithm for mining meta-rule guided single-variable strong ML-association rules using the p-predicate testing technique.

Example 3.2 We examine the derivation of the multiple-level strong association rules for query (q_1) of Example 2.1.

1. The same as Step 1 and Step 2 of Example 3.1.

2. Large p-predicate-sets at multiple concept levels, i.e., $\mathcal{L}[1,p]$, $\mathcal{L}[2,p]$, ..., $\mathcal{L}[max\_l,p]$, are derived based on the large 1-predicate sets derived at previous step. This skips the generation of the large 2-predicate tables of Example 3.1 and generates only the large 3-predicate tables as Table 4.

3. The rules in the form of meta-rules are generated from the large p-predicate-sets at each concept level based on the specified confidence threshold at this level. This generates the same rule table as Table 5. □

Algorithm 3.2 (Direct p-predicate testing)
Meta-rule guided mining of single-variable strong ML-association rules using the direct p-predicate derivation technique.

Input: The same as Algorithm 3.1.

Output: The same as Algorithm 3.1.

Method: A top-down, progressively deepening process which collects large p predicate sets at multiple concept levels as follows.

1. The same as Step 1 and 2 of Algorithm 3.1.

2. Derive the large p-predicate-set tables at each concept level from level 1 to max_l, i.e., derive $\mathcal{L}[l,p]$, for $l = 1, \ldots, max\_l$, where p is the total number of predicates in the meta-rule.

   Note that a set of p predicates is large at level l if (1) each of its component 1-predicates is large at level l, and (2) the support of the p predicates at level l is no less than $minsup[l]$.
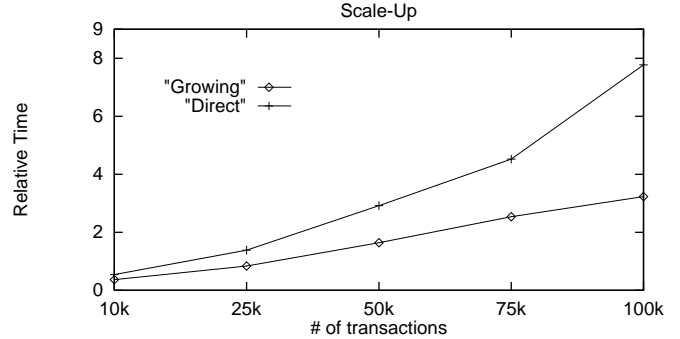

Figure 1: Scale up of the Algorithms

3. For each concept level l, generate the rules in the form of meta-rules from the large p-predicate set tables $\mathcal{L}[l,p]$ if the confidence of the rule is no less than $minconf[l]$, the specified confidence threshold at this level. □

3.3 A performance comparison of the two algorithms

We implemented the two algorithms on a SUN SparcStation5 with 32MB main memory. A synthetical database is used to test the algorithms. The database has five attributes each of which has 100 values at the primitive level. The values are organized into a concept hierarchy with four levels. The numbers of higher level(nonprimitive) nodes in the hierarchy are 1, 5 and 20 at level 1, 2, 3 respectively. Since there are only one node at the level 1, it is treated as virtual level and does not join the computation. The meta-rule we used has the form: $A(t,x) \land B(t,y) \rightarrow C(t,z)$. The minimal confidences are 50% at all levels.

First, we test the scale-up properties of the two algorithms. They are tested on the database with the number of tuples from 10,000 to 100,000. The minimal supports are (4%, 1%, 0.2%) at the level 2, 3 and 4. The performance data are shown in Figure 1. As we can see, both algorithms scale up well. Algorithm 3.1 has better scale-up behavior since the overhead of computing $L[l,k]$ for small k weights less and less as the database size grows.

We then compare the performance of the algorithms under different minimal supports. Figure 2 shows the execution times of both algorithms with different minimal supports. The database size is fixed at 10,000 tuples. The minimal supports used are: T1(6%, 1%, 0.5%), T2(4%, 1%, 0.1%), T3(4%, 0.5%, 0.1%), T4(2%, 0.5%, 0.1%), and T5(2%, 0.5%, 0.05%). When the minimal supports decrease, the execution times increase since the filter is weaker. We find that Algorithm 3.1 is sensitive to the minimal supports since it uses them to cut small patterns at each iteration. On the other hand, Algorithm 3.2 is not so sensitive to the change. Algorithm 3.1 outperforms Algorithm 3.2 when the minimal sup-
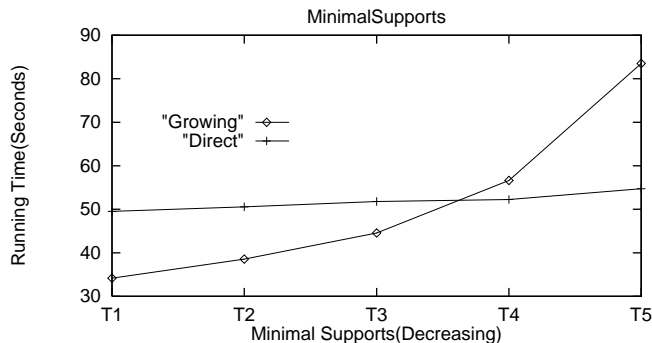
Figure 2: Different Minimal Supports

ports are large(so the filter is strong) while Algorithm 3.2 outperforms Algorithm 3.1 when the filter is not so strong. Generally, we feel Algorithm 3.1 should be tried for most reasonable support thresholds. Algorithm 3.2 is a good candidate when lots of details are interested.

4 Meta-rule guided mining of multiple-variable rules

Now we examine the meta-rule guided mining of multiple-variable rules. Since a multiple-variable association rule presents relationships among several relations, a join of these relations should be performed in the data collection step based on the join relationship explicitly expressed in the meta-rules.

Taking query (q_2) in Example 2.1 as an example, we analyze the data mining process as follows.

Example 4.1 The meta-rule presented in query (q_2) of Example 2.1 contains three predicates: $major(s, x)$, $P(c, y)$, and $Q(s, c, z)$. The first predicate is from the attribute *major* of the relation *student*, the second is a property in relevance to the relation *course* because it contains one variable from *course*, and the third is a property in relevance to the relation *grading* since it contains two variables, each from *student* and *course*, respectively.

The data mining process is to discover the relationships in relevance to three relations: *student*, *course*, and *grading*. It is necessary to perform a join of the three relations. Since only one predicate $major(s, x)$ is from the relation *student*, only the attribute *major* in the relation *student* is retained in the joined relation. Therefore, the joined relation should have the following schema.

s_c_g (sno, major, cno, title, dept, instructor, semester, grade)

The possible instantiations of the two candidate predicates P and Q should be: $P \in \{title, dept\}$, and $Q \in \{instructor, semester, grade\}$. Moreover, since *title* is unique in the relation *course*, which is similar to

the behavior of the key *cno*, the predicate P in the meta-rule can only be instantiated to *dept*. Therefore, the data mining process is essentially to find multiple-level association rules in relevance to the following three properties: (1) $major(s, x)$, (2) $dept(c, y)$, and (3) one of the following three predicates: $instructor(s, c, z)$, $semester(s, c, z)$, $grade(s, c, z)$.

Except for the restriction on the instantiation of predicate variables, the data mining method are like that of mining single-variable association rules.   □

5 Discussion

This section discusses some closely-related issues on meta-rule guided mining of multiple-level association rules, including meta-rule-guided mining of mixed-level rules and variations of constraints on the forms of meta-rules.

5.1 Meta-rule-guided mining of mixed-level rules

In the method developed in the last section, it is assumed that the concepts of the predicates in the discovered rules are lined up among different predicates according to the levels of their concept hierarchies. For example, major "Science" is lined up with gpa "Excellent" and birth_place "Western Canada", whereas major "CS" is lined up with gpa "3.8_3.9" and birth_place "N. Burnaby", etc. However, it may not be the case in practical applications. It could be desirable to line up major "CS" with gpa "Excellent" and birth_place "British Columbia", etc. That is, it is often necessary to link concepts among different predicates at multiple levels of hierarchies for effective knowledge mining.

Interestingly, the method studied in the last two sections can be modified minorly to accommodate this flexible data mining requirement. For example, Algorithm 3.1 can be modified as the following for mining rules across multiple concept levels. At the third step, the candidate large 2-predicate-sets will enclose the pairs of two large 1-predicate-sets at any concept levels instead of pairing only those at the same concept levels.

5.2 Variations of constraints on the forms of meta-rules

In our previous discussion, there has been another constraint on the possible forms of meta-rules: there are no repetitive predicate variables in the meta-rule, and all the predicates in an instantiated rule will be different.

Although this restriction may cover a large number of applications, there are applications which would like to study the association relationships involving the same predicates. For example, one may like to find the gen-

eral association relationships among the courses taken by the same student. Such a query could be presented and examined in the following example.

Example 5.1 In the university database of Example 2.1, one may like to find the association relationships among the courses taken by the same student. The query can be presented as follows.

(q_4) : discover rules in the form of
$P(s : S, c_1 : C, x_1) \wedge P(s, c_2 : C, x_2) \to P(s, c_3 : C, x_3)$
from student S, grading G, course C

The system may find some meaningful rules like the following.

$$grade(s, \text{``}CMPT100\text{''}, \text{``}Excellent\text{''}) \wedge$$
$$grade(s, \text{``}MATH100\text{''}, \text{``}Excellent\text{''})$$
$$\to grade(s, \text{``}CMPT300\text{''}, \text{``}A\text{''}) \quad (82\%)$$

Note in this case the data mining process can be viewed as a similar process of mining association rules in transaction databases [1]. This is because the relational table can be compressed into a table consisting of two fields: (1) a set of distinct students, each corresponding to a transaction identifier in a transaction database, and (2) a set of corresponding grading records associated with each student, each corresponding a set of data items processed by that transaction. Thus the transaction-based data mining algorithms developed in previous studies [2, 6] can be applied in the efficient processing of association relationships. However, the previously developed transaction-based association rule mining algorithms still need to be modified to accommodate more complicated queries.

6   Conclusions

We studied the meta-rule guided mining of multiple-level association rules in large relational databases. Meta-rule guided mining of multiple-level association rules provides syntactic constraints on the desired rule forms to be discovered, which leads to the constrained and progressive mining of refined knowledge from data and thus has interesting applications for knowledge discovery in large databases.

A top-down progressive deepening data mining technique is developed for rule-guided mining of multiple-level association rules, which extends the mining of multiple-level association rules mining algorithms to rule-guided mining of association rules. Two algorithms, , have been proposed and tested against synthesized databases, and their performance study shows that different algorithms may have the best performance for different distributions of data.

Related issues, including concept hierarchy handling, methods for mining flexible multiple-level association rules, and adaptation to difference mining requests are also discussed in the paper. Our study shows that meta-rule guided mining of multiple-level association rules from databases has wide applications and efficient algorithms can be developed for discovery of interesting and strong such rules in large databases.

Integration of rule-guided mining and the mining of multiple-level knowledge rules poses many issues for further investigation. For example, meta-rule guided mining of multiple-level sequential patterns, the patterns containing aggregation functions, etc. are interesting topics for future study.

References

[1] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In *Proc. 1993 ACM-SIGMOD Int. Conf. Management of Data*, pages 207–216, Washington, D.C., May 1993.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proc. 1994 Int. Conf. Very Large Data Bases*, pages 487–499, Santiago, Chile, September 1994.

[3] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. *Advances in Knowledge Discovery and Data Mining*. AAAI/MIT Press, 1995.

[4] J. Han, Y. Cai, and N. Cercone. Data-driven discovery of quantitative rules in relational databases. *IEEE Trans. Knowledge and Data Engineering*, 5:29–40, 1993.

[5] J. Han and Y. Fu. Dynamic generation and refinement of concept hierarchies for knowledge discovery in databases. In *Proc. AAAI'94 Workshop on Knowledge Discovery in Databases (KDD '94)*, pages 157–168, Seattle, WA, July 1994.

[6] J. Han and Y. Fu. Discovery of multiple-level association rules from large databases. In *Proc. 1995 Int. Conf. Very Large Data Bases*, Zurich, Switzerland, Sept. 1995.

[7] J. Han and Y. Fu. Exploration of the power of attribute-oriented induction in data mining. In U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*. AAAI/MIT Press, 1995.

[8] M. Klemettinen, H. Mannila, P. Ronkainen, H. Toivonen, and A. I. Verkamo. Finding interesting rules from large sets of discovered association rules. In *Proc. 3rd Int'l Conf. on Information and Knowledge Management*, pages 401–408, Gaithersburg, Maryland, Nov. 1994.

[9] G. Piatetsky-Shapiro and W. J. Frawley. *Knowledge Discovery in Databases*. AAAI/MIT Press, 1991.

[10] W. Shen, K. Ong, B. Mitbander, and C. Zaniolo. Meta-queries for data mining. In U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*. AAAI/MIT Press, 1995.