Int j simul model 13 (2014) 4, 458-471
ISSN 1726-4529 Original scientific paper

ASSOCIATION RULES ALGORITHM AND ITS APPLICATION IN THE MAINTENANCE OF THE TUNNEL

Xu, W.-X.[*]; Fan, Y.-H.[*] & Zhang, J.-X.[**]
[*] School of Traffic and Transportation, Beijing Jiaotong University, Beijing, China
[**] School of Languages and Communication Studies, Beijing Jiaotong University, Beijing, China
E-Mail: wxxu@bjtu.edu.cn, yhfan@bjtu.edu.cn, jxzhang@bjtu.edu.cn

Abstract
In this paper, effective data mining methods are adopted for the tunnel management information system to deal with safety issue data and work out the relationship among these safety issues in order to estimate risk, establish intelligent decision support, provide basis of governance for railway maintenance departments and remedy the defects of the existed management information systems. In view of the bottleneck of Apriori algorithm, two new algorithms are proposed in this paper. The first is AprioriN algorithm based on arrays, which converts the operation on database to the operation on memory via coding. The second is a high performance association rule mining algorithm based on FP-tree, which accelerates the speed of traverse itemsets by adding an extra data structure. During the second scan of the database, a matrix is generated to save frequent 2-itemsets when the basic FP-tree is created. This paper attempts the improved algorithms to improve the efficiency.
(Received, processed and accepted by the Chinese Representative Office.)

Key Words: Data Mining, Association Rule, Frequent Itemsets, FP-Tree, Tunnel Safety Issue

1. INTRODUCTION

The first railway tunnel in China was constructed in 1888 and after more than one century of construction, there were 7153 railway tunnels in mainland China with a total length of 4601.836 km by the end of 2011. However, a very stunning fact is that most of them are suffering from safety defects originated from low designing standards, ill-considered situation, and poor quality of construction and shortage of maintenance management. Some of the defects endanger the traffic safety directly, while some others potentially, and the railway sectors annually invest a lot of manpower, funds and other resources for the maintenance and remediation of the tunnel safety, but they still couldn't make a fundamental turn for better conditions of the tunnel and the amount of data are idled. To ensure a safe operation of tunnels, equipment management department ought to master all the information which may manifest insecurity and take appropriate measures to eliminate the hidden dangers of the tunnel as far as possible. For a long time in the past years, the collection of tunnel insecurity information mainly relies on human eye-viewing, the measuring tapes and other simple and crude devices. Data management is no more than live recording data to fill in routine reports. Thus are the results to follow: an incomplete understanding of the security information, poor management, shortage of risk knowledge and inadequate prevention against accidents. Such status certainly calls for reforms in information management and renovation in the study of methodologies.

As one of the human-dominated computer systems, management information system utilizes computer hardware, software and other office equipment to collect, transmit, store, process and maintain information in terms of supporting the operation, management and decision-making. The existed railway tunnel management information systems have already achieved the goal of managing raw data, images and text reports dynamically, with establishment of models and improvement of visual degree of data. However, most of these systems only focus on a specific class of factors or engineering conditions. The studies on

safety issue mechanism and relationship of safety issue factors are quite neglected in the observation of the potential rule of safety issues via data mining, and as a result, the reliability and operability of renovation are not ideal. In terms of the management information system of railway tunnel lining, this article endeavours to offer a systematic analysis, with a summary of the features of the railway tunnel safety problems, a revelation of the hidden relationship among the data of the existed safety issues in railway tunnel through mining historical data, and provides strong decision support for the routine maintenance of the railway tunnel, and for safety defects prediction and remediation. The association rule method for data mining is a powerful tool from which an effective data mining method can be evolved to carry out excavation of safety issue data and map out the relationship among the safety issues so as to estimate risks, establish the intelligent decision support, provide bases of harness for railway detection and maintenance departments, and remedy the defects of the existed management information systems.

The association rule method for data mining can be conducted to sort out the potential relationship of the tunnel safety issues through the mining of historical data [1]. If safety problems can be effectively detected and governed, the probability of the occurrence of other relevant problems will be greatly reduced, and so will the treatment cost. These rules play an important role in the development of safety defects detection and safety problem management system. With decision support and data mining for guidance, we integrate data acquisition, analysis, and control; implement monitoring and unified decision support method, and build functions of the healthy status of the railway tunnel decision support system platform connected with data warehouse and data mining technology. We use this platform to complete the function of collecting tunnel safety data, processing them and assessing the healthy status of the tunnel with them. Also we can facilitate a control over the basic parameter data of tunnel, the data of tunnel safety defects, safety problem management situation and association between safety issues in real-time, and therefore, prompt and accurate mastery of the changes of the security status in existed tunnel can be achieved, the right countermeasures be taken, hidden dangers eliminated and the safety of operators tunnel ensured [2].

Many algorithms for mining association rules from transaction databases have been proposed since Apriori algorithm was presented [3]. However, these algorithms were based on Apriori algorithm so they needed to generate and test candidate itemsets, which means a very high computational cost [4]. In order to overcome the bottleneck of the Apriori algorithm and mine association rules without candidate itemsets, FP-growth algorithm was proposed. The algorithm was used to compress a database into a tree structure which shows a better performance than Apriori. However, FP-growth algorithm consumes more memory and performs poorly with long pattern data sets [5]. Recently, many researchers have been trying to find efficient methods to improve Apriori algorithm and FP-growth algorithm [6]. In this paper, we propose two algorithms named AprioriN algorithm and FP-growthN algorithm, which can efficiently overcome the disadvantages and whose computational results have confirmed their good performances. The FP-growthN algorithm is applied to the railway tunnel management system to achieve the safety assessment, establish the intelligent decision support and offer the excavation basis for railway reconnaissance.

2. THE ALGORITHM ARCHITECTURES OVERVIEW

Data mining is one of the research areas for the huge amounts of database and decision-making information, of which the main purpose is to develop the relevant methods, theories and tools to extract useful and interesting knowledge from large amounts of data [7], which is potentially useful and the process of which includes the data definition, data quality improvement, data preparation, data mining model, data mining and knowledge assessment.

Association rule mining is one of the data mining methods [8], which is an important class of knowledge representation to reveal implicit relationships among the items in large number of transactions, manifest the relevance in large data sets and establish the association rule of them [9].

2.1 Current state of Apriori algorithm

Let $I = \{ I_1,  , \ldots, I_m \}$ be an itemset. Let D be a transactional database. Each transaction T is a subset of I, belongs to D and contains an identification number TID. Let A be an itemset. Transaction T contains A if and only if $A \subseteq T$. An association rule is an implication of the form $A \Rightarrow B$, where $A \subset I$, $B \subset I$, and $A \cap B = \emptyset$. Rule: $A \Rightarrow B$ is established in a transactional database whose support is s and confidence is c. S is the number of transactions in D that contain $A \cup B$. The specific description is as follows.

Support $(A \Rightarrow B) = P(A \cup B)$

Confidence $(A \Rightarrow B) = P(B|A)$

Association rules that satisfy both the minimum support threshold and minimum confidence threshold are called strong. The minimum support threshold can be shorthanded for *min_sup*. The minimum confidence threshold can be shorthanded for *min_conf*. The threshold values are between 0 % and 100 %.

In general, association rule mining can be viewed as a two-step process:

Step 1: Find all frequent itemsets: By definition, each of these itemsets will occur at least as frequently as a predetermined minimum support count, *min_sup*.

Step 2: Generate strong association rules from the frequent itemsets: by definition, these rules must satisfy minimum support and minimum confidence.

2.2 Current state of FP-growth algorithm

As is seen, in many cases the Apriori candidate generate-and-test method significantly reduces the size of candidate sets and leads to good performance. However, it suffers from two nontrivial costs:

1) It may need to generate a huge number of candidate sets.

2) It may need to repeatedly scan the database and check a large set of candidate by patterns. It is costly to go over each transaction to determine the support of the candidate itemsets [10].

The FP-growth adopts a divide-and-conquer strategy. First, it compresses the database representing frequent items into a frequent-pattern tree, or FP-tree, which retains the association information of the itemsets. Then it divides the compressed database into a set of conditional databases, each associated with one frequent item. FP-growth algorithm transforms the problem of searching for long frequent patterns to searching for shorter ones recursively and then concatenating the suffix. It uses the least frequent items as a suffix, offering good selectivity. The method substantially reduces the searching costs.

Table I: A sample database.

TID	Items	Frequent items/PDB
100	f, a, c, d, g, i, m, p	f, c, a, m, p
101	a, b, c, f, l, o	f, c, a, b, o
102	b, f ,h, j, m, p	f, b, m, p
103	b, c, k, m, o, s	c, b, m, o
104	a, f, c, e, l, n, o, p	f, c, a, o, p

The process is to construct a compressed tree structure to mine frequent pattern effectively.

Case: transactional database *D* and minimum support threshold *min_sup* =3

First, a FP-tree is constructed as follows.

1) It consists of the root of the tree, a group of the root node of the prefix sub-tree and a header table.

2) The prefix sub-tree consists of item *ID*, support count and node-link. The support count expresses the number of the transaction. Node-link points to the next same item node (not existed is null).

3) An item header table consists of item *ID* and the chain of node-links. The chain of node-link points to the first node.

The FP-tree is mined as follows. Start from each frequent length-1 pattern (as an initial suffix pattern), construct its conditional pattern base (a "subdatabase" which consists of the set of prefix paths in the FP-tree co-occurring with the suffix pattern), then construct its (conditional) FP-tree, and perform mining recursively on such a tree. The pattern growth is achieved by the concatenation of the suffix pattern with the frequent patterns generated from a conditional FP-tree.

Mining of the FP-tree is summarized in Table II and detailed as follows based on FP-growth algorithm. Since the initial parameters are null and there is no single path, in reverse search to find the last *p* in the item header table. *P* is in the three branches of the tree: <f:4, c:3, a:3, m:1, p:1>, <f:4, c:3, a:3, o:1, p:1> and <c:1, b:1, m:1, p:1>. First path refers to the first time of <f, c, a, m, p> in database. Although itemset <f, c, a> appears three times and <f> appears four times, they only appear one time with *P*. So we only have statistics the prefix path <fcam:1>. Similarly to the above analysis, the second and third prefix paths respectively are <fcao:1> and <cbm:1>. They form the p's conditional base. Therefore, FP-tree only contains <c:3>, not <f, a, m, b, o>. The support of itemsets is smaller than the minimum support threshold 3. So the frequent itemset is {(cp:3)}. The process of mining by parity of reasoning and results are summarized in Table II.

Table II: Create conditional pattern base to mine FP-tree.

Item	Conditional Pattern Base	Conditional FP-tree	Frequent Pattern
p	{(fcam:1), (fcao:1), (cbm:1)}	<c:3>	cp:3
o	{(fcab:1), (fca:1), (cbm:1)}	<c:3>	co:3
m	{(fca:1), (fb:1), (cb:1)}	Ø	Ø
b	{(fca:1), (f:1), (c:1)}	Ø	Ø
a	{(fc:3)}	<fc:3>	fa:3, ca:3, fca:3
c	{(f:3)}	<f:3>	fc:3
f	Ø	Ø	Ø

2.3 Apriori algorithm issues

There are two bottlenecks of traditional Apriori algorithm: first, it consumes I/O in scanning the data frequently one by one in database; second, it consumes the CPU to compare the records in string pattern to match the algorithm.

2.4 FP-growth algorithm issues

The study of the FP-growth algorithm shows that it must scan the database three times to generate a FP-tree. First, it generates frequent 1-itemset *F* and forms the frequent itemset table. Second, it sorts all frequent items in the transaction database. Third, it scans the sorted frequent items to generate the FP-tree. The bottleneck of the algorithm is concentrated in the

third traverse as it needs to determine each frequent item in each transaction and determine how to insert into FP-tree.

3. ALGORITHM OPTIMIZATION FOR ASSOCIATION RULES

Two bottlenecks of the traditional Apriori algorithm are as follows:

First, scanning the data frequently one by one in database is at the cost of I/O;

Second, it consumes CPU to compare the records in string pattern to match the algorithm.

In fact, itemsets and the size of the amount of data are often staggering in data mining. Consequently, Apriori algorithm is not suitable for multi-itemsets and multi-data mining.

The study of FP-growth algorithm shows that it must scan the database three times when it is generating FP-tree. First, it is used to generate frequent 1-itemset F and forms the frequent itemset table. Second, it sorts all frequent items in the transaction database. Third, it scans the sorted frequent items to generate FP-tree. The bottleneck of the algorithm is concentrated in the third traverse. Because it needs to determine the each frequent item in the each transaction and determine how to insert into FP-tree. So the overhead of the system will become very expensive and efficiency will be greatly reduced [11].

3.1 AprioriN algorithm

Recently, many researchers have tried to find efficient methods and technologies to improve Apriori algorithm [12]. Some algorithms increase the speed, but reduce the accuracy. Some improved algorithms such as hash tree technology only concern about the process. Although division data improved algorithm only scans the database two times, its threshold is not reasonable. Almost all improved algorithms neglect the researching data preprocessing.

In this paper, we propose a new algorithm named AprioriN algorithm, which encodes the original database. AprioriN algorithm based on arrays converts the operation on database into the operation on memory via coding. It is a great improvement and the experimental result shows that new algorithm is very effective.

3.1.1 Improved algorithm idea

1) Data preprocessing: Scan the database to find the frequent 1-itemset; then set frequent 1-itemset simple coding. Encoding: $T_i = \sum 2^t$ where t is the transactional number, for example: A transactional data {I1,I4,I5}={1,0,0,1,1} can be encoded as $2^0 + 2^3 + 2^4 = 25$. Let the largest transactions be 5. The value of m is 5. The original sample database and its encoding are shown in Table III.

Table III: The original sample database and its encoding.

Sequences	Values	Code	Coding
T1	I1,I2	1,1,0,0,0	3
T2	I1,I3	1,0,1,0,0	5
T3	I1,I4,I5	1,0,0,1,1	25

2) After getting frequent 1-itemset L_1, find l_1 and l_2 randomly in the frequent 1-itemset. If $(l_1[1]=l_2[1]) \wedge ... \wedge (l_1[k\text{-}2]=l_2[k\text{-}2]) \wedge (l_1[k\text{-}1]<l_2[k\text{-}1])$, then connect $c=l_1 \oplus l_2$, and encode c. If it is not generated candidate itemset, it should be given up. Finally, it generates candidate 2-itemsets C_2.

3) The third step is to sort the data in memory to determine the candidate c-itemsets contained in each decoded itemsets and get frequent 2-itemsets by judge and merger.

4) Find frequent k-itemsets l_k in the same way.

3.1.2 Test of AprioriN algorithm

In the railway tunnel lining condition monitoring data management system, it runs Apriori algorithm and AprioriN algorithm, produces a brief description of AprioriN algorithm and mines the data which comes from the autumn routine inspection.

The operation environment of this system: CPU frequency is 1.80 GHz, 1 GB memory laptop, operating system is Microsoft Windows XP Professional SP2. Choose .NET language as complier. The test database is autumn examination statistics.

The data which comes from the autumn routine inspection in this system are raw data, and not convenient to be mined directly. Before data mining, it is necessary to do previous work of data preparing. Data preprocessing is an important step in data mining, which can handle missing data and clean dirty data in order to improve the reliability of the results and complete data conversion. There are 15 categories of tunnel safety issues in Table IV.

Table IV: Tunnel safety issue coding table.

Tunnel safety issue name	**Encoded**	**TID**
Lining deformation or movement	S0101	1
Lining cracks or dislocation	S0102	2
Lining crushing	S0103	3
Lining corrosion	S0201	4
Leakage	S0202	5
Frost damage	S0301	6
Drainage facilities damaged	S0302	7
Hole Yang slope landslide rock fall	S0303	8
Insufficient clearance	S0401	9
Operating ventilation	S0402	10
Poor lighting	S0403	11
Tunnel monolithic bed deformation damage	S0501	12
Tunnel invert deformation damage	S0502	13
Tunnel bed damage	S0503	14
Insufficient strength of lining	S0602	15

1) To deal with the data using SQL statement.

```
select x.tunnelname, x.CRACKITEMID
    from tunnel_item  x
        where ((select count (*)
         from tunnel_item  y
       where x.tunnelname = y.tunnelname and checkyear='2009') >= 2)
order by tunnelname, crackgrade;
```

2) Then based on the generated data, set the minimum support: $min\_sup = 0.2$ to generate the frequent 1-itemset.

The mining results of association rule are as follows.

Table V: Frequent 1-itemset.

item 1	**cnt**
S0201	2
S0202	5
S0503	2

3) Prune and delete not only the items which don't exist in frequent itemsets, but also delete the items whose id number can't generate the frequent 2-itemsets. Then connect to generate the frequent 2-itemsets. The mining result of association rule is as follows.

Table VI: Frequent 2-itemsets.

item 1	item 2	cnt
S0201	S0202	2

4) Continue loop, and the frequent 3-itemsets is empty.

5) According to the mining result, set *min_sup* = 50 %. The mining results of association rule are as follows.

Table VII: The mining result of association rule.

SN	Association rule	Confidence
1	2⇒5	conf(2|5) = 887/1498*100 % = 59.2 %
2	5⇒9	conf(5|9) = 1215/2403*100 % = 50.6 %
3	11⇒5	conf(11|5) = 536/821*100 % = 62.3 %
4	14⇒9	conf(14|7) = 561/821*100 % = 68.3 %
5	2, 9⇒5	conf(2|9, 5) = 479/707*100 % = 67.8 %
6	2, 5⇒9	conf(2|5, 9) = 479/887*100 % = 54.0 %

(1) 2⇒5

This association rule shows that if the lining cracks or dislocation happens, the possibility of leakage is 59.2 %.

(2) 5⇒9

The leakage can cause the insufficient clearance. However, this rule cannot be seen intuitively, thus we should pay attention to it.

(3) 11⇒5

It refers that if poor lighting exists, it will cause leakage largely and affect the traffic safety. However, this rule cannot be seen intuitively, so it should arouse the attention of the parties concerned.

(4) 14⇒9

The tunnel bed damage can cause the happening of the insufficient clearance. This association rule is in accord with our mind.

(5) 2, 9⇒5

It can be found that when lining cracking and dislocation and insufficient clearance appear at the same time, leakage is also a great possibility.

(6) 2, 5⇒9

Similarly, it can be found that when lining cracking and dislocation and leakage appear at the same time, insufficient clearance is also a great possibility. Fig. 1 compares the Apriori algorithm with AprioriN algorithm in time efficiency of mining data.

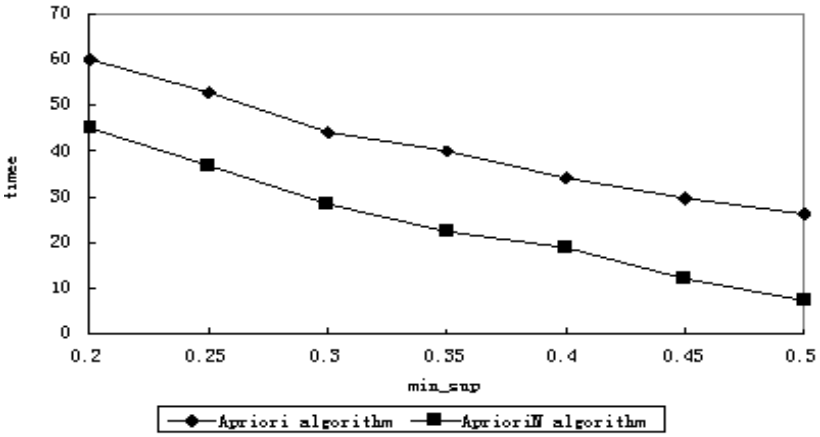

Figure 1: Comparison of two algorithm's running time.

As AprioriN algorithm modifies just the efficiency of the algorithm, not the structure of it, thus the frequent items are the same as the result of the traditional algorithm. However, the improved algorithm has also bottlenecks. When meeting large sum of data, such as more than ten million, the algorithm efficiency will sharply decline because of the computer memory problem. Fig. 2 compares the Apriori algorithm with AprioriN algorithm and marks the turning point of the bottlenecks of the latter.

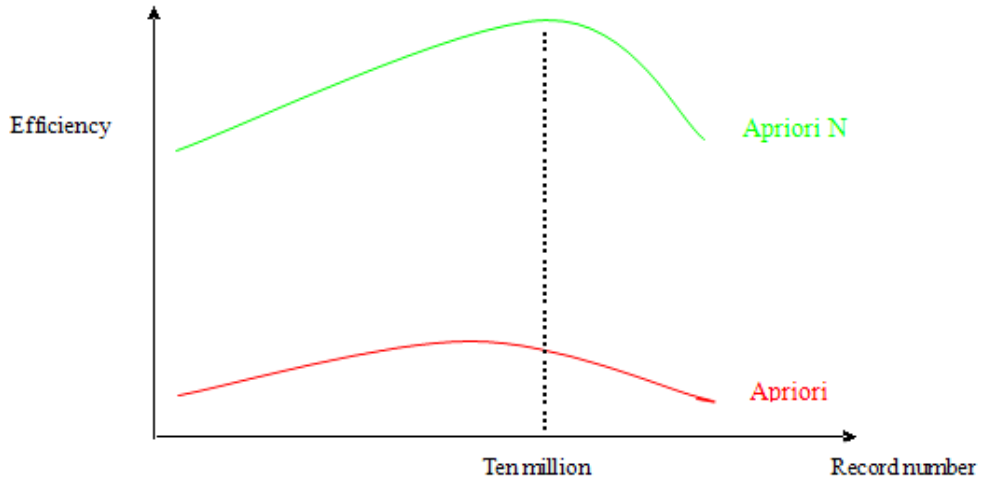

Figure 2: The bottlenecks of two algorithm's running time.

Although the new algorithm's time complexity has increased, the computational efficiency of it has been greatly improved; the efficiency of the memory operations is much higher than the database I/O operating. The experimental result shows that the efficiency of new algorithm is 3.5 times faster than that of traditional algorithm.

3.2 FP-growthN algorithm

The FP-growth method transforms the problem of finding long frequent patterns to searching for shorter ones recursively and then concatenating the suffix [13]. It uses the least frequent items as a suffix, offering good selectivity. The method substantially reduces the search costs.

However, studies on the performance of the FP-growth method shows that it takes 80 % of CPU to scan the FP-tree.

FP-growthN algorithm accelerates the traverse speed of itemsets through adding an extra data structure for finding frequent itemsets without candidate generation [14].

Method: Adding an additional data structure to accelerate the speed of traverse the itemsets and scan the data second time to generate FP-tree ($T_{\emptyset}$) and matrix $A_{\emptyset}$ which is used to save the frequent 2-itemsets.

3.2.1 Improved algorithm processing

FP-growthN: Mine frequent pattern in T_{β} and matrix A_{β}.

Input: FP-tree; transaction database D; minimum support threshold *min_sup*.

Output: The complete set of frequent patterns.

Method: Call FP-growthN (T, null)

Procedure FP-growthN (T, α)

(1) If T contains a single path P then

(2) For each combination (denoted as β) of the nodes in the path P;

(3) Generate pattern $\alpha \cup \beta$ with the support = a_i.support;

(4) Else for each a_i in the header of Tree {

(5) Construct $\beta = a_i \cup \alpha$ and support = a_i.support;

(6) if *T*.array is not NULL
(7) construct β's conditional pattern base and then β's conditional FP-tree Tree;
(8) else create a new header
(9) construct β's conditional pattern T_β and matrix A_β
(10) if $T_\beta \neq \Phi$ then
(11) call FP-growth(T_β, β); }

3.2.2 FP-growthN algorithm characteristics
The bottleneck of traditional FP-growth algorithm is the third scanning, since each transaction in the database for each frequent item in the process of the constructed FP-tree must be judged one by one to decide how to join the FP-tree. Scanning FP-tree consumes 80 % of CPU. By adding matrix to reduce the scanning time, the new algorithm will increase the mining speed.

3.2.3 Test FP-growthN algorithm
Similarly, in the railway tunnel lining condition monitoring data management system, it runs FP-growthN algorithm and offers a brief description of FP-growthN algorithm.

1) Scan the database to find the frequent 1-itemset. The set of frequent items is sorted in descending order of frequent 1-itemset support count and pruned to remove the frequent 1-itemset that is not frequent simultaneously. Run the program to produce the basic FP-tree and Matrix $A_\emptyset$. The result is that the tree has 119 nodes and the depth is 8.

(*x.x.x*) *y*:*z* (ref to *y*1:*z*1) refers to the node whose support is *z* in itemset *y* of branch (*x.x.x*), and the pointer of it points to the node whose support is *z*1 in *y*1 itemset $A_\emptyset$. The matrix is shown in Fig. 3.

2	ϕ	28	147	51	46	130	42	53
4	28	ϕ	56	15	15	58	14	23
5	147	56	ϕ	63	72	203	66	79
7	51	15	63	ϕ	22	65	23	32
8	46	15	72	22	ϕ	63	20	17
9	130	58	203	65	63	ϕ	64	75
11	42	14	66	23	20	64	ϕ	26
14	53	23	79	32	17	75	26	ϕ
	2	4	5	7	8	9	11	14

$A_\emptyset$

Figure 3: Matrix $A_\emptyset$.

2) Based on $T_\emptyset$ and $A_\emptyset$, we recursively get $T_{\{\beta\}}$ and $A_{\{\beta\}}$. The final frequent itemsets:{5}= 271{9}=257{9,5}=203{2}=187{2,5}=147{2,9}=130{2,9,5}=102{14}=106{14,5}=79{14,9} =75{8}=97{8,5}=72{7}=92{11}=81{4}=72. frequent 1-itemset:{5}{9}{2}{14}{8}{7}{11} {4}; frequent 2-itemsets: {2,5}{9,5} {2,9}{14,5}{14,9}{8,5}; frequent 3-itemsets: {2,9,5}.

3) Calculate the support:

In Table VIII, the analyses of the results are as follows.

Table VIII: The mining result of association rule.

SN	Association rule	Confidence
1	{2, 5}	P(5|2) = P(5,2)/P(2) = 147/187 = 79 %
2	{9, 5}	P(9|5) = P(9,5)/P(5) = 203 /271 = 74.5 %
3	{2, 9}	P(9|2) = P(9,2)/P(2) = 130/187 = 69.5 %
4	{14, 5}	P(5|14) = P(14,5)/P(14) = 79/106 = 74.5 %
5	{14, 9}	P(9|14) = P(14,9)/P(14) = 75/106 = 71 %
6	{8, 5}	P(5|8) = P(8,5)/P(8) = 72/97 = 74 %
7	{2, 9, 5}	P(5|2,9) = 102/130 = 78 %

(1) {2, 5}:

Because 79 % > 70 %, it can be seen that if lining cracks or dislocation appears, the leakage will appear. That is, if lining cracks or dislocation can cause leakage, it will affect the traffic safety. It coincides with our mind, which means that the association rule has practical significance.

(2) {9, 5}:

P(5|9) = P(5,9)/P(9) = 203/257 = 89 %. Because of 89 % >> 70 %, tunnel may exist insufficient clearance which would lead to the leakage of water. The consequence will affect the traffic safety. However, this rule cannot be seen intuitively, so it should arouse the attention of the parties concerned. Similarly, because of 74.5 % > 70 %, the leakage of water also leads to insufficient clearance, they affect each other.

(3) {2, 9}:

Because the percentage doesn't reach the minimum support threshold, 2 (Lining cracks or dislocation) has nothing to do with 9 (Insufficient clearance).

(4) {14, 5}:

It can be seen that 14 (Tunnel bed damage) can cause the happening of 5 (Seepage).

(5) {14, 9}:

It can be seen that 14 (Tunnel bed damage) can cause the happening of 9 (Insufficient clearance).

(6) {8, 5}:

It can be seen that 8 (Hole Yang slope landslide rock fall) can case the happening of 5 (Seepage).

(7) {2, 9, 5}:

It can be found that when 2 (Lining cracks or dislocation) and 9 (Insufficient clearance) appear at the same time, the possibility of 5 (Seepage) is large.

It is can be seen that most of the tunnel safety issues occur in groups from the above mining results. If one safety issue happens, it may cause another or several safety issues. In Fig. 4, it compares the FP-growth algorithm with FP-growthN algorithm.

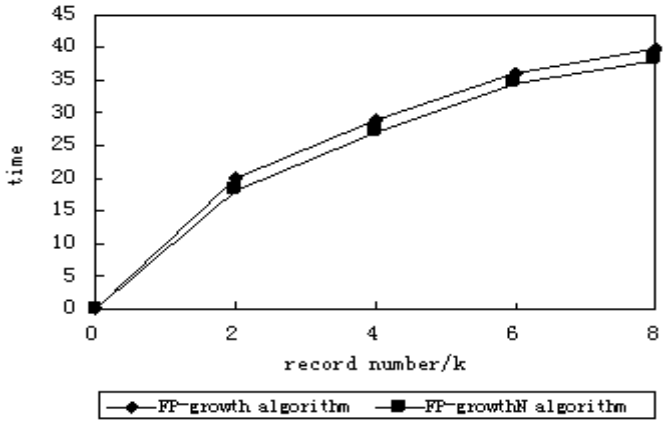

Figure 4: Comparison of two algorithm's running time.

4. THE APPLICATION OF FP-GROWTHN ALGORITHM

In this paper, we present a high performance association rule mining algorithm based on FP-tree and FP-growthN algorithm, which accelerates the traverse speed of itemsets through adding an extra data structure. During the second-time scan of the database, a matrix that can be used for saving frequent 2-itemsets is generated while the basic FP-tree is being created. Then the improved algorithm is applied to the railway tunnel management information system, which selects the detection tunnel, inputs weights of safety issue factors, runs the FP-

growthN algorithm, analyses association among safety issues, carries out the timely descriptions and feedbacks and establishes intelligent decision support.

With the implementation of the leap-forward development strategy and greatly improved speed, higher demands on the state of tunnel safety and security information management level are thus put forward with the application of the advanced technologies, the controls over the security changes and the appropriate measures to eliminate the hidden dangers and to ensure the traffic safety.

With monitoring decision support system, supervisor can have real-time access to important information of the tunnel maintenance; do real-time or periodic monitoring for deformation of lining, concrete stress, the groundwater level as well as other issues. Supervisor can also analyse the tunnel characteristics with mining historical data, analyse the relationship of the safety issues and put forward more rational prevention and treatment programs. In most cases, the system will periodically compare deformation data with the prescribed safety standards. When the deformation exceeds a certain level, the system will automatically send alarming signals to the maintenance engineer.

Compared with the traditional crude statistics and query, we have improved the association rule to work out the relationship of tunnel safety issues and find the hidden knowledge which can't be obtained by statistics and query. The application of data mining for the railway tunnel safety problem prevention and management department provides an effective decision support. This process is shown in Fig. 5.

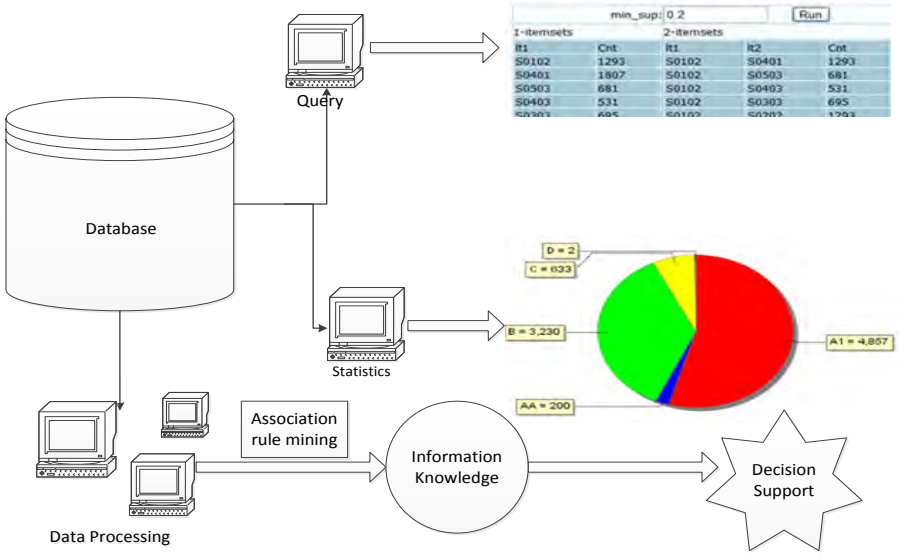

Figure 5: Data mining processing.

The system is generally divided into seven modules: the tunnel file module, safety issue module, data transmission extraction, assessment and decision support, monitoring module, data mining, and system management module. This system mainly takes video surveillance and then gets the information into the system through the long-distance transmission. The data mining module is further divided into data processing sub-module and association mining sub-module. The association rule method for data mining can help figure out significant relationships among the tunnel safety issues through mining historical data. The system management module is divided into the administrative user module and ordinary user module, with the former enabled to modify the data from the system and maintain the system, and the latter only to register, modify personal information and query basically.

With focuses on safety issue module, assessment and decision-making module and data mining module, this paper tests algorithms and elaborates upon the significance of the algorithm in the system.

In the railway tunnel lining state test data management system, safety issue module mainly records the safety issues information and gives timely description and feedback on the safety status quo. So the module must be dynamic and real-time. In addition to providing information query, it also includes data mining capability which provides safety issue association rule analysis and mines the association of the various historical safety issues. Statistical analysis of the degradation project interface is shown in Fig. 6.

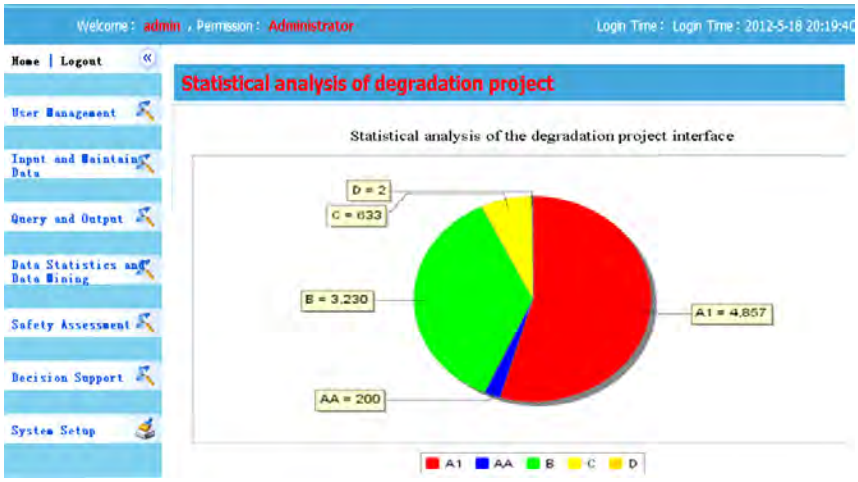

Figure 6: Statistical analysis of the degradation project.

The data mining module, which is divided into data processing sub-module and association mining sub-module, mainly performs the association rule mining analysis for the historical data.

The processes of data mining are as follows.

1) Set the minimum support = 20 %, and run the FP-growth algorithm to get the frequent itemsets. The interface is shown in Fig. 7.

2) In the tunnel safety assessment function, the system is carried out to achieve stratified analysis. The system distinguishes different lining, specifies different types of evaluation, supports the evaluation of multi-section and integrates the results of the score of each section to obtain overall conclusions of the safety of the tunnel. FP-growthN Algorithm is applied to the system to test the safety issue status of the Nan wu tai tunnels.

3) Set weights, run the algorithm to detect the tunnel status. Fig. 8 shows the interface.

4) If the condition of the tunnel is safety, the system doesn't deal with it. Otherwise, the system will prompt the types of safety issues and the corresponding measures.

It1	Cnt
S0102	1293
S0401	1807
S0503	681
S0403	531
S0303	695
S0202	2234
S0302	488
S0201	266

It1	It2	Cnt
S0102	S0401	1293
S0102	S0503	681
S0102	S0403	531
S0102	S0303	695
S0102	S0202	1293
S0102	S0302	488
S0102	S0201	266
S0401	S0503	681
S0401	S0403	531
S0401	S0303	695
S0401	S0202	1807
S0401	S0302	488
S0401	S0201	266
S0503	S0403	531
S0503	S0303	681
S0503	S0202	681
S0503	S0302	488
S0503	S0201	266
S0403	S0303	531
S0403	S0202	531
S0403	S0302	488
S0403	S0201	266

It1	It2	It3	Cnt
S0102	S0503	S0403	531
S0102	S0503	S0303	681
S0102	S0503	S0202	681
S0102	S0503	S0302	488
S0102	S0503	S0201	266
S0102	S0403	S0303	531
S0102	S0403	S0202	531
S0102	S0403	S0302	488
S0102	S0403	S0201	266
S0102	S0303	S0202	695
S0102	S0303	S0302	488
S0102	S0303	S0201	266
S0102	S0302	S0201	266
S0401	S0503	S0403	531
S0401	S0503	S0303	681
S0401	S0503	S0202	681
S0401	S0503	S0302	488
S0401	S0503	S0201	266
S0401	S0403	S0303	531
S0401	S0403	S0202	531
S0401	S0403	S0302	488
S0401	S0403	S0201	266

Figure 7: The frequent itemsets of *min_sup* = 20%.

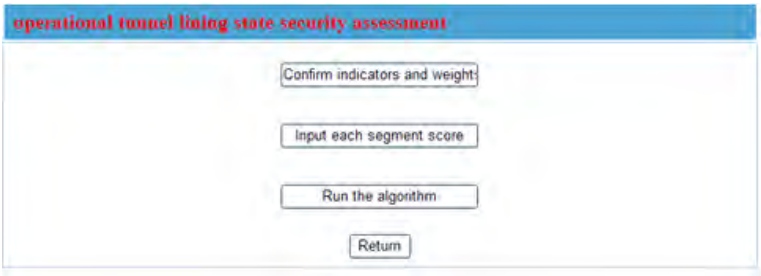

Figure 8: Set up weight and run the algorithm.

In the past, the great bulk of historical data of constructional defects failed to provide effective decision making support for the work of governing and controlling. People used to control the safety issue based on severity level rather than those based on the association of the safety issues. Therefore, the work of governing and controlling was ineffective and inefficient. Our studies have demonstrated that the association rule method for data mining can provide effective instruction for the controlling and governing of the tunnel safety issues through mining historical data. On the basis of the association of safety issues, the railway sector will greatly improve its effectiveness and efficiency of governance.

5. CONCLUSIONS AND FUTURE WORK

It is well known that the railway sectors collect the tunnel safety indicators every year, and great amount of data have been accumulated so far. However, these data are either mostly idled or fail to play an effective role in the prediction and prevention of safety issues.

It is true that the existed railway tunnel management information systems have already achieved the goal of managing raw data, images and text reports dynamically, establishing model and improving visual degree of data; however, most systems only focus on a specific class of safety issues or engineering conditions, and the awareness of the safety issue mechanism and relationship among these safety factors are poor and the potential safety issue discipline has not been mined. Thus, the reliability and operability of renovation measures are not ideal.

In the light of the system elaborated in this paper, the association rule method for data mining can provide effective instruction for the controlling and governing of the tunnel safety issues through mining historical data. In this way, the relationship between different safety issues can be figured out and therefore, if one safety issue can be effectively tackled, the occurrence probability of other relevant ones will be greatly reduced. These association rules practically will play important roles in constituting the detection criterion and controlling the tunnel damage for railway departments. What has been discussed can be summed up in the following two aspects.

1) The systems we have established in our study can be adopted effectively to find the underlying relationship of different safety issues and extract their characteristics.

2) Based on the system and the mining results, real-time monitoring and the procedure of safety assessment can be provided, and counter-measures be taken in case that the railway tunnel can’t comply with safety standard.

Although the performance of our algorithm (AprioriN algorithm) is greatly improved, placing the encoded data will consume the memory for such a large amount of data, as over millions. Therefore, our future work will focus on the distributed computing. Specifically, when the amount of data is too large, such computing method can split data to increase efficiency.

Increasing additional matrix can greatly enhance the performance of the FP-Growth algorithm in dealing with sparse data, but if the data is too dense, adding additional matrix

will only exhaust the computer's memory and CPU consumption and it can't accelerate the speed of mining. Therefore, we need to find enhancement algorithms to deal with the dense data in the future. As geologic conditions differ greatly all over China, the diversity of the railway tunnel structures is also enormous. Association rule can be implemented not only to detect the relationship among the various safety issues but also extract the common characteristics of tunnel geological conditions.

6. ACKNOWLEDGEMENTS

This research has been sponsored and supported by the National Natural Science Foundation of China (61272029) and National Key Technology R&D Program (2009BAG12A10).

REFERENCES

[1] Wang, X. & Liu, M. G. (2010). Association rules mining, *Economic Research Guide*, Vol. 2010, No. 11, 198-199

[2] Xu, W. X.; Xu, L.; Liu, X. M.; Jones, J. D. (2008). A new approach to decision-making with key constraint and its application in enterprise information systems, *Enterprise Information Systems*, Vol. 2, No. 3, 287-308, doi:10.1080/17517570802302341

[3] Lee, N.; Jung, J. J. (2012). Profit-based association rule mining from commercial transactions, *Information – An International Interdisciplinary Journal*, Vol. 15, No. 8, 3469-3476

[4] Abdullah, Z.; Herawan, T.; Deris, M. M. (2012). Mining highly-correlated least association rules using scalable trie-based algorithm, *Journal of the Chinese Institute of Engineers*, Vol. 35, No. 5, 547-554, doi:10.1080/02533839.2012.679064

[5] Achar, A.; Laxman, S.; Sastry, P. S. (2012). A unified view of the apriori-based algorithms for frequent episode discovery, *Knowledge and Information Systems*, Vol. 31, No. 2, 223-250, doi:10.1007/s10115-011-0408-2

[6] Chen, K.; Zhang, L.-J.; Li, S.; Ke, W. (2011). Research on association rules parallel algorithm based on FP-growth, Liu, C.; Chang, J.; Yang, A. (Eds.) *Information Computing and Applications*, Springer-Verlag, Berlin, 249-256, doi:10.1007/978-3-642-27452-7_33

[7] Jiang, H.; Zhao, Y.-Y.; Dong, X.-J. (2008). Mining positive and negative weighted association rules from frequent itemsets based on internet, *International Symposium on Computational Intelligence and Design*, Wuhan, 242-245, doi:10.1109/ISCID.2008.172

[8] Abdullah, Z.; Herawan, T.; Deris, M. M. (2010). Mining significant least association rules using fast SLP-growth algorithm, Kim, T. H.; Adeli, H. (Eds.) *Advances in Computer Science and Information Technology*, Springer-Verlag, Berlin, 324-336, doi:10.1007/978-3-642-13577-4_28

[9] Amin, G. R.; Gattoufi, S.; & Seraji, E. R. (2011). A maximum discrimination DEA method for ranking association rules in data mining, *International Journal of Computer Mathematics*, Vol. 88, No. 11, 2233-2245, doi:10.1080/00207160.2010.543457

[10] Totad, S. G.; Geeta, R. B.; Prasad Reddy, P. V. G. D. (2012). Batch incremental processing for FP-tree construction using FP-Growth algorithm, *Knowledge and Information Systems*, Vol. 33, No. 2, 475-490, doi:10.1007/s10115-012-0514-9

[11] Lin, K.-C.; Liao, I.-E.; Chen, Z.-S. (2011). An improved frequent pattern growth method for mining association rules, *Expert Systems with Applications*, Vol. 38, No. 5, 5154-5161, doi:10.1016/j.eswa.2010.10.047

[12] Mustafa, M. D.; Nabila, N. F.; Evans, D. J.; Saman, M. Y.; Mamat, A. (2006). Association rules on significant rare data using second support, *International Journal of Computer Mathematics*, Vol. 83, No. 1, 69-80, doi:10.1080/00207160500113330

[13] Liu, Y.-M.; Guan, Y. (2009). Application in market basket research based on FP-Growth algorithm, *2009 WRI World Congress on Computer Science and Information Engineering*, Vol. 4, 112-115

[14] Li, H.; Wang, Y.; Zhang, D.; Zhang, M.; Chang, E. Y. (2008). Pfp: Parallel FP-Growth for query recommendation, *Proceedings of the 2008 ACM Conference on Recommender Systems*, 107-114