Application of Incremental Updating Association Mining Algorithm in Geological Disasters System

Wang Jianguo
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, China
e-mail: wjg_xit@126.com

Zhu Ying
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, China
e-mail: 642454565@qq.com

Abstract—**Aiming at the problems of low efficiency, low cost of time and space, this paper proposes an algorithm to update the association mining of the inverted index tree. The algorithm combines the inverted index technology with the tree structure. When the data in the database is continuously updated, it can scan only the newly added part of the database, without having to scan the original database to count the number of transaction items. The optimal threshold predicted by Newton's interpolation formula is compared with this frequency to get frequent item sets. Then, the confidence level is calculated for the combinations of different item sets in frequent item sets, and the correlation rules are obtained, and the correlation analysis of the rules is carried out to obtain a more realistic association rule. The inverted index tree updating association mining algorithm was applied to the data analysis of geological hazards monitoring system. One year data record of rainfall, groundwater level, soil water content and topography data was selected as the experimental data set. Compared with the IUAR algorithm, it is found that the inverted index tree updating association mining algorithm has some improvements in memory consumption and efficiency. The experimental results show that when the minimum support of IUAR algorithm remains unchanged, the number of transaction records is the same as the amount of new data, and Inverted Index Tree Incremental Updating Association Mining Algorithm takes less than 2/5 of the IUAR algorithm. When the number of transaction records and the amount of new data remain unchanged and IUAR algorithm support changes, the Inverted Index Tree Incremental Updating Association Mining Algorithm memory consumption is much smaller than IUAR algorithm. In the process of experiment, according to the results of the Inverted Index Tree Incremental Updating Association Mining Algorithm, the association rules are obtained and the correlation is judged. The strong association rules are used to set the alarm threshold of the geological disaster monitoring system.**

Keywords-Inverted Index Tree; Geological Disaster System; Frequent Item Sets; Association Rules

I. INTRODUCTION

Geological disasters are geological phenomena that cause serious harm or potential threat to human lives and property. Human activities can affect the occurrence of geological disasters and the extent of their damage, but they can not be completely eliminated or prevented. In addition to earthquakes, volcanoes, tsunamis and other sudden geological disasters can cause unmanageable destructive disaster to humans. Some slowly changing geological disasters such as landslides, land subsidence and ground fissures will also bring huge losses to the lives and property of the people, the economic development in cities and areas. Geological disasters have seriously affected the sustainable development of society and affected social stability. Therefore, geological disaster data is an important basic resource related to national economy and the people's livelihood, because the data contains a lot of useful information. But understanding and relying on these data to make scientific decisions is beyond human capacity. How to make full use of geological exploration data to make relevant prediction and scientific decision-making has become one of the concerns of production decision makers.

Association Rule Mining[1] is one of the most explored, most commonly used data mining technology. This method is one of the most active research areas in the field of data mining. It mainly helps to discover the implicit and valuable relationships between data items in massive databases to guide decision makers in various fields such as commerce Strategic analysis. Researching association rule algorithms in big data technology environment is a very important and challenging research topic. Considering that the data in the database of geological disasters always change constantly, the incremental association rule updating techniques have been proposed to effectively maintain the association rules of updated data. The technology should have some characteristics:

- The association rules should vary with the data;
- The rule updating should avoid dealing with the old data again, as much as possible to use the previous processing results;
- Updating maintenance methods should be applied to a variety of occasions as much as possible. FUP algorithm is iterative update algorithm based on Apriori algorithm.

In order to solve the problem of incremental updating mining of association rules, Cheung and other scholars proposed a fast update algorithm (FUP)[2]. Later, domestic scholars such as Zhu Yuquan proposed a method FUKFIA[3] for rapidly updating frequent item sets based on the FUP algorithm. They define a new set of frequent items that reduce the number of database scans to a certain extent.

Inevitably, the above incremental updating algorithm of association rules, like the Apriori algorithm, requires layer-by-layer traversal to generate frequent item sets. Therefore, there is also the problem of overhead in processing databases and generating huge candidate item sets.FUP2 algorithm[4] is based on the FUP algorithm to improve, put forward an improved algorithm for transaction records in the constantly updated, correct and delete operations. Feng Yucai and other scholars put forward incremental update association mining algorithm IUA and PIUA[5]. In the algorithm, splicing and pruning techniques are used to solve the generation of candidate item sets. When the transaction database is unchanged and the minimum support threshold is changed. CATS-tree algorithm[6], IUAR algorithm[7] are through a variety of methods to reduce the number of scanning the original database to achieve incremental update association rules maintenance issues.

To sum up, the FUP algorithm and the FUP2 algorithm are used to maintain the incremental updating association mining when the minimum support threshold and the minimum confidence threshold do not change, and the transaction records are continuously updated. IUA algorithm, PIUA algorithm and CATS-tree algorithm handle the maintenance of incremental updating association mining when the transaction record data does not change, the minimum support threshold and the minimum confidence threshold change. The IUAR algorithm is used to solve the problem of maintaining and updating the association mining when the transaction database is continuously updated and the minimum support threshold is changed. The basic idea of the algorithm is to obtain the extended set of candidate frequent items by reducing the support degree. When accessing the updated database, the association rules are incrementally updated by constantly updating the candidate frequent item sets. Although this algorithm has been greatly optimized in terms of the large number of candidate item sets, there are still some shortcomings that it is necessary to retrieve the original transaction database multiple times and produce a large number of candidate item sets. So far, there has been relatively little research on incremental updates in the context of big data environments when both the transaction database and the minimum support threshold are changed. Therefore, it is very necessary to find an incremental update mining method that can effectively solve such problems.

In this paper, we propose an incremental update mining algorithm for inverted index tree, which is applied to geological disaster monitoring system. Firstly, the frequent item sets are excavated according to the database, and then the association rules are obtained and analyzed. Finally, the association rules are applied to the early warning work.

II. Inverted index and inverted index tree

A. *Inverted index*

The inverted index is the most commonly used data structure in the information retrieval system. In the index, each index item consists of the attribute value and the location information that appears,<Key, storage address>. At the time of querying, you can get all the documents that contain the keyword at once, so the retrieval efficiency is higher than the forward index. For example, Table1 is a partial transaction log of the groundwater table (derived from the original transaction database of the Geological Disaster Monitoring System), and Figure 1 is the inverted index map (IIP)[8] of TABLE I.

TABLE I. Groundwater table in geological disasters database (mm)

ID	Groundwater level
0	48
1	52
2	46
3	43
4	42
5	52
6	53
7	47
8	42
9	40
10	39
11	54

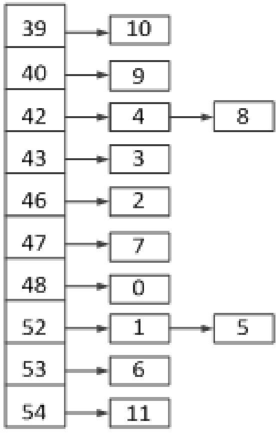

Figure 1. TABLE.I corresponds to the inverted index map(IIP)

B. *Inverted index tree based on B+ tree implementation*

B+ tree is the deformation of the B-tree. A m-order B+ tree[9] should meet the following characteristics:

- The number of keywords per node is equal to the number of children. The keywords of all non-lowest inner nodes are the largest keywords on the corresponding sub-tree, and the bottom node contains all the keywords;
- Branch nodes can be placed (m-1) keyword, leaf nodes can put m keywords;
- All leaf nodes are in the same layer of the number structure and do not contain any information. Thus, the tree height of the tree is balanced.

B+ tree is a commonly used index mechanism in the database and a one-dimensional data index structure design[10].

As shown in Figure 2, a B+ tree consisting of the divided character string is m=3.

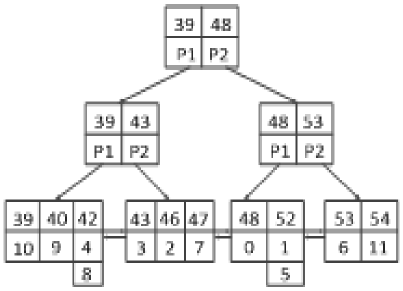

Figure 2. Order three of the B+ tree

III. Algorithm of Incremental Updating Rule Mining for Inverted Index Tree

The above traditional frequent item sets mining algorithm has disadvantages when generating association rules mining will generate a large number of candidate item sets and repeat retrieval processing of the original database. In this paper, we implement the inverted index tree based on the characteristics of B+ tree and put forward the incremental update association mining algorithm of inverted index tree to deal with the association rules efficiently.

A. Basic ideas

- Statistics in the database transaction items, get transaction items set.
- constructs the inverted index tree (IITree) based on the B+ tree creation method.

The bottom-most leaf node of IITree contains all the item sets. The frequent item sets are obtained by comparing the number of items with the minimum support, and other infrequent item sets remain in their leaf nodes to ensure that future data updating become frequent nodes. The different item sets in the frequent item sets are combined and their confidences are calculated. When the confidence is greater than the minimum confidence, the item set combination is the association rule.

Compared with the previous incremental updating correlation mining algorithm, Inverted Index Tree Incremental Updating Association mining algorithm introduces B+ tree structure, as well as the database settings. When adding new data, we only need to retrieve the frequent item sets that deal with the new part of the data without the need to re-retrieve the entire database.

The algorithm takes advantage of the B+ tree's balanced tree properties. The leaf nodes at the bottom of the tree contain all the keywords of the whole tree, and they are linked together like a doubly linked list. The inverted index[9] is realized based on the B+ tree .

B. Set the threshold

The minimum support threshold and the minimum confidence threshold will change with different user needs and database updates. When the threshold is set too low, the more rules that are excavated, the lower the usefulness of the rules. Conversely, when the threshold is set too high, there are few rules for mining, so some useful rules will be lost. Therefore, setting the appropriate threshold is very important when dealing with incremental databases.

- When mining association rules for the first time, setting the minimum support threshold is a trial and error. Select a small part of data sets randomly from the entire database to be excavated, set initial support thresholds and confidence thresholds according to the user's requirements or experience, and obtain n number of association rules. Compare n with the number of association rules the user expects m. If $n/n^{'} \geq d$, it is considered that the threshold set by the user is smaller, so that the excess number of rules dug up is expected. We should increase the support threshold by a certain value and rerun it. If $b < n/n^{'} < d$,it is considered that the user is substantially satisfied with the result of the association rule mined at this support threshold. If $n/n^{'} < b$,it is considered that the set threshold is too large, some important rules may be lost, and then a slightly smaller support threshold is selected to re-excavate the algorithm.
- When the transaction database is updated, it is possible that the previously set threshold is no longer applicable, so the threshold needs to be reset. Based on the support threshold, confidence threshold, the number of association rules output by the algorithm at the last time and the current mining targets, the Newton interpolation formula is used to predict the support threshold that should be adopted currently, which makes the mining association rules more effective .

C. Algorithm implementation

In order to achieve the inverted index[11] with the B+ tree, Inverted index tree incremental update association mining algorithm is proposed.

1) The algorithm steps are briefly described as follows:

a) Traverse the inverted index map, get the item set. Based on the data of groundwater level, soil moisture content, rainfall and topography in the database of geological disaster monitoring system, inverted index maps were established. Traversing the index graph to get the item set, and then by the B+ tree to build inverted index tree.

b) Get all frequent item sets based on the generated IITree. In the B+ tree, the bottom leaf node contains all the keywords. Then, the confidence level is calculated for the combinations of different item sets in frequent item sets, and the correlation rules are obtained, and the correlation analysis of the rules is carried out to obtain a more realistic association rule.

c) When the transaction database update records, in accordance with the above steps to retrieve some of the new data processing, the item set inserted IITree. Add a keyword to the leaf node at the bottom of the IITree. If the number of keywords contained in the node does not exceed M, the insertion is successful. Otherwise, the node needs to be split. The number of keywords included in the two new nodes after splitting should not be less than (M/2 + 1). If the parent node of the node has enough space (M/2 ~ M-1), the parent node should contain the smallest keyword in the right node. Otherwise, the parent node is split and a new keyword is added to the parent of the parent node, followed by a higher order until the new root node is generated (1 ~ M-1).

d) After the database is updated, repeat the Step2 operation and then generate the association rules.

Incremental update is illustrated by groundwater level data. TABLE II is the database updating data, Figure 3 is the entire database corresponding to the II Tree.

TABLE II. UPDATE THE RECORD DATA (MM)

ID	12	13
Groundwater level	57	41

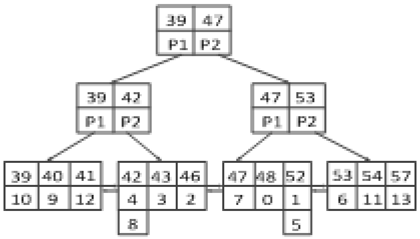

Figure 3. Inverted index tree of updated data

2) Algorithm flow chart shown in Figure 4:

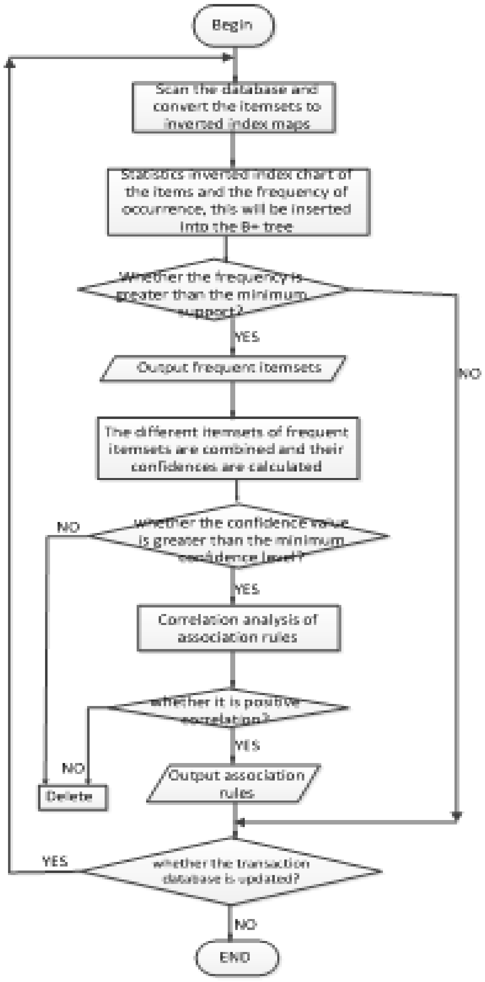
Begin
Scan the database and convert the itemsets to inverted index maps
Statistics inverted index chart of the items and the frequency of occurrence, this will be inserted into the B+ tree
Whether the frequency is greater than the minimum support?
YES
NO
Output frequent itemsets
The different itemsets of frequent itemsets are combined and their confidences are calculated
whether the confidence value is greater than the minimum confidence level?
NO
YES
Correlation analysis of association rules
whether it is positive correlation?
YES
NO
Delete
Output association rules
YES
whether the transaction database is updated?
NO
END

Figure 4. Flow chart of incremental index updating algorithm of inverted index tree mining

D. Correlation Analysis of Association Rules

Although, association rules have metrics of interest such as support and confidence. However, association rules may include causal association as well as random association or even negative correlation. Here's an example to illustrate:

In a supermarket database system, the customer's product purchase information is recorded. Of the 10000 purchases, 8000 of them included bread, 6000 had cookies, 4800 had both breads and biscuits. If you set a minimum support of 30% and a minimum confidence of 40%, then can get the following association rules:

Rule 1: Buy bread ⇒ Buy biscuits $\{\sup port = 48\%, confidence = 60\%\}$

In reality buying bread and buying biscuits may be negative because buying bread will reduce the number of people buying biscuits. At the same time, consider the following negative correlation rules:

Rule 2: Buy bread ⇒ Do not buy cookies $\{\sup port = 32\%, confidence = 40\%\}$

In a sense, the second rule is more realistic. Thus, under given threshold conditions, two contradictory rules are obtained.

It can be seen from the above examples that judging the true meaning of association rules can not be based solely on the measure of support and confidence, but rather on a comprehensive examination of the data set. To do this, put forward some other methods such as chi-square statistics or correlation analysis[12]. The core idea of these methods is to measure the correlation between data items. Chi-square statistics calculation formula is as follows(χ^2):

$$\chi^2(A,B) = \frac{[P(A)*P(B) - P(A \cup B)]^2}{P(A)*P(B)} \tag{1}$$

If chi-square statistics is zero, then there is no dependency between the data item A and the data item B, and they are independent of each other. Otherwise, the data items are interdependent. Relevance calculations more clearly show that this dependence is mutual promotion or mutual restraint. Correlation is calculated as follows:

$$corr(A,B) = \frac{p(A \cup B)}{P(A)*P(B)} \tag{2}$$

If $corr(A,B)$ is equal to 1, then data item A and data item B are independent; if $corr(A,B)$ is greater than 1, data item A and data item B are positively correlated; if $corr(A,B)$ is less than 1, then data item A and data item B are negative Related.

Support-Confidence Frame Theory is not perfect: Some rules are of no practical value, even if both support and confidence are high. The association rule $A \to B$ does not give the user information whether A and B are constructive or counterproductive. Relevance analysis of association rules is to overcome this deficiency, allowing users to rationally view the association rules. Therefore, after the association rules of the geological disaster monitoring system database are excavated, the correlation analysis should be carried out to ensure the practicability of this rule. If it is determined that the rule is positively correlated, the value of the rule is used as the alarm threshold of the geological hazard monitoring system. In the future of new data acquisition, if the conditions of this rule, then the alarm. People can prevent it in advance.

IV. EXPERIMENT RESULTS AND ANALYSIS

The algorithm uses the record of rainfall, groundwater level, soil water content and topography data of Shang Nan County in Shang Luo City in the geological disaster monitoring system of last year as the experimental data set. Use C language in Win7, dual-core 2.3GHZ CPU, 4GB memory on the PC for simulation.

A. Comparison of IITree algorithm and IUAR algorithm

1) The analysis of time complexity of the algorithm:

a) When the data is updated, only in the database to scan the updated data;

b) When building an inverted index tree, scan the tree structure once and insert the new item set. Analysis shows that when the minimum support is constant, the execution time of the algorithm is related to the amount of data updated each time. To extract a small number of experimental samples from the data set, the minimum support for controlling IUAR algorithm is unchanged (0.1), increase the amount of updated data in turn. Record the experiment time of IUAR algorithm and IITree algorithm respectively, time comparison shown in Figure 5:

TABLE III. IUAR, IITREE ALGORITHM TO RUN THE EXPERIMENTAL TIME(S)

Data Set		Time
Twenty thousand	IUAR	43
<No new ones>	IITree	18
forty thousand	IUAR	96
<Add 20,000>	IITree	33
Eighty thousand	IUAR	176
<Add 40,000>	IITree	84
One hundred thousand	IUAR	402
<Add 20,000>	IITree	134

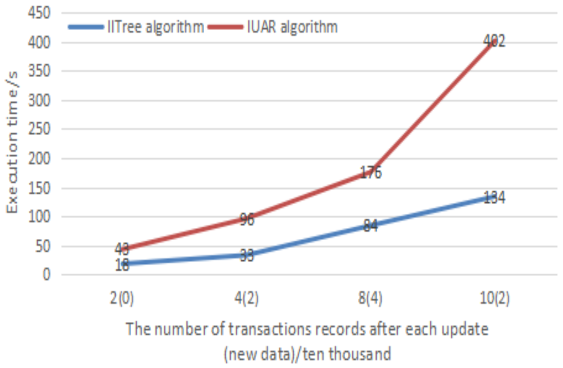

Figure 5. Algorithms time comparison for IUAR and IITree

As shown in Figure 5, when the minimum support is constant, the execution time of the IUAR algorithm increases rapidly but the ones of the IITree algorithm grows more slowly when the amount of updated data increases. In the same amount of data, IUAR algorithm takes more time than IITree.

2) The analysis of spatial complexity of the algorithm:

a) In the inverted index map, only the updated data is stored, so the size of the memory space is related to the amount of updated data;

b) In the IITree algorithm, the frequent item sets determined by the minimum support are stored, so the memory space is associated with the minimum support. This experiment mainly studies the effect of minimum support on memory usage. When the minimum support of IUAR algorithm is changed from 0.2 to 0.6, the data samples and increments remain unchanged. 800 new records have been added to the raw groundwater level data set as test samples. The memory usage of IUAR algorithm and IITree algorithm is compared according to the experimental results.

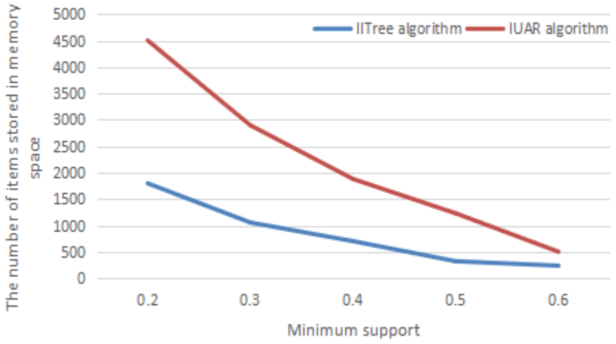

Figure 6. Memory usage comparison

Figure 6 shows that the smaller the minimum support, the more the number of item sets produced, the greater the memory footprint. In the case of support change, the IUAR algorithm updates the candidate set with the change of the support degree, saves the candidate item set and the frequent item set in the memory space. The IITree algorithm does not generate the candidate item set in the change of the support degree, so the occupied memory space by IUAR algorithm is greater than the ones by IITree algorithm.

B. Application of IITree Algorithm in Geological Disaster Monitoring System

If the relationship between the table properties are Boolean attributes, then mining rules from this relational table are Boolean rules. The problem now is that geological disaster monitoring system databases are numerical data. The quantitative attributes must be dealt with in a necessary way so that the mining of quantitative rules can be transformed into the mining of Boolean rules. The main strategy is to divide the range of the number attribute into intervals, and to decompose a quantity attribute into several Boolean attributes. In order to reduce the computational workload, the original data are standardized and divided into different sections. The data are grouped according to the sections and the frequency is recorded in the IITree. Then the frequent item sets can be excavated. The frequent item sets are divided according to the average value and divided into low value area and high value area respectively. Data of groundwater level, rainfall, soil moisture content and ground deformation in the past year were selected from the database of geological disasters and the association rules were excavated. Select some of the data for analysis, as shown in TABLE IV:

TABLE IV. Disaster monitoring data to Boolean data conversion table

Num	**Ground water level**		**Rainfall**		**Soil moisture content**		**Ground deformation**	
ID	***L*** ***G1***	***H*** ***G2***	***L*** ***R1***	***H*** ***R2***	***L*** ***S1***	***H*** ***S2***	***S*** ***D1***	***B*** ***D2***
1	0	1	1	0	0	1	0	1
2	0	1	1	0	1	0	1	0
3	0	1	0	1	0	1	0	1
4	1	0	0	1	1	0	0	1
5	0	1	1	0	0	1	0	1
6	1	0	0	1	1	0	0	1
7	0	1	1	0	0	1	1	0
8	1	0	0	1	1	0	1	0
9	1	0	0	1	0	1	0	1
10	1	0	1	0	1	0	0	1
...	...	...	...	...	...	...	...	...

According to the data of geological hazard monitoring system, the association rule mining is carried out and a series of rules are obtained. Some rules are as follows:

Rule 1: If G2=1 and R1=1 and S2=1 then D2=1

Rule 2: If G1=1 and R2=1 and S1=1 then D2=1

Rule 3: If G2=1 and R2=1 and S2=1 then D2=1

Rule 4: If G2=1 and R1=1 and S2=1 then D2=1

Rule 5: If G1=1 and R2=1 and S1=1 then D1=1

Rule 6: If G1=1 and R1=1 and S1=1 then D2=1

After analyzing the above rules, we can get:

- In the case of high groundwater tables, heavy rainfall, and high soil moisture levels, large-scale deformation of the ground is promoted (according to rules 1 and 3);
- Under conditions of heavy rainfall, the deformation of the ground may also be induced, even if the groundwater level and soil moisture are not high (according to Rule 2);
- When the groundwater level and soil moisture content is high, it is possible to promote the occurrence of ground deformation (according to Rule 4);
- When the groundwater level is low, the soil moisture content is low and the rainfall is very little, the ground will be deformed due to dryness (according to Rule 6).

In summary, when the data of the local water table, rainfall, soil moisture content and ground deformation reach the value of the association rules, further analysis can be used as the warning threshold of landslide or ground fissure.

V. Conclusion

In this paper, An algorithm of Inverted Index Tree Incremental Updating Association Mining (IITree) is proposed. The algorithm is effectively implemented when the database is updated, without having to scan the original database. The new data will be inserted into the original B+ tree to get frequent item sets. Experiment results show that the IITree algorithm consumes less than 2/5 of the IUAR algorithm when the number of transactions and the amount of new data are the same, which improves the efficiency of data processing. When the minimum support of IUAR algorithm changes, IITree algorithm takes up less memory than IUAR algorithm. The application of IITree algorithm to the data analysis of geological disaster monitoring system has some improvements in efficiency and memory usage and can be better applied to the early warning of the system.

REFERENCES

[1] Rakesh Agrawal,Tomasz Imieliński,Arun Swami. Mining association rules between sets of items in large databases[J]. ACM SIGMOD Record,1993,22(2).

[2] David W.Cheung, Jiawei Han. Maintenance of Discovered Association Rules in Large Databases: An Incremental Updating Technique. In Proc of the Twelfth International Conference on Data Engineering,1996.USA: IEEE, 1996(3):106-114.

[3] Zhu Yuquan, Sun Zhizhu, Zhao Chuan Shen.Quickly update frequent itemsets[J].Computer Research and Development,2003(01):94-99.

[4] David W L,Cheung S,Lee D,et al.A general incremental technique for maintaining discovered association rules[C].In Proceedings of the Fifth International Conference On Database Systems For Advanced Applications,Melbourne,Australia,1997(3):185-194.

[5] FENG Yucai, FENG Jianlin. Incremental Updating Algorithm for Association Rules[J] .Journal of Software, 1998,9 (4): 301.

[6] William Cheung, Osmar R.Zaiane. Incremental Mining of Frequent Patterns without Candidate Generation or Support Constraint .In Proc of the Seventh International Conference on Database Engineering and Applications Symposium,2003. USA:IEEE, 2003(7):111-116.

[7] Gao Feng, Xie Jianying.Discover the incremental update algorithm of association rules[J].Computer Engineering,2000(12):49-50+112.

[8] LI Wen,HONG Qin,TENG Zhongjian,SHI Zhaoying. An inverted index based on B+ tree [J]. Computer Knowledge and Technology, 2011,3 (8): 1720.

[9] HU Yanbo,ZHONG Jun. Based on clustering B + tree database index optimization algorithm [J]. Computer Applications, 2013,33 (9): 2474.

[10] Roh Hongchan, Kim Woo-Cheol, Kim Seungwoo, et, al. A B-Tree index extension to enhance response time and the life cycle of flash memory[J]. Information Sciences,2009,179:3136.

[11] WANG Yingqiang,SHI Yongsheng. Application of B+ Tree in Database Index[J]. Journal of Yangtze University, 2008,3 (5): 233.

[12] Chen Xiaojiang, Huang Zhang Chan.Numerical Analysis[M].Beijing: Science Press,2010:30-34.