Optimized Apriori Algorithm Description and Application in Vehicle Fault Phenomenon Correlation Analysis

Hao Kang¹, and Hailong Zhao²

¹Sergeant School of Army Academy of Armored Forces, ChangChun, China
²Sergeant School of Army Academy of Armored Forces, ChangChun, China

Abstract. This paper proposes an optimized Apriori data correlation analysis algorithm and describes it, and evaluates the results of the model in the correlation analysis of vehicle fault phenomena. The algorithm reduces the load of I/O interface and improves the computational efficiency by optimizing the number of frequent itemset and pruning in the data mining process. The algorithm is applied to the correlation analysis of common faults in vehicle engine system. The optimized traversal method enables it to quickly find frequent patterns, correlations and causal structures among fault data sets, and detect potential correlations among faults, which achieves better application effect.

1 Introduction

Data mining is an automatic or semi-automated way to explore and analyze large amounts of data to find valuable patterns, and then use the rules embodied in the patterns to improve the efficiency and quality of data application. Data mining technology has gradually become a powerful tool for the current information industry, social related application industry to obtain useful knowledge and information.

Correlation analysis is a simple and practical technique for data mining, which is a way to discover the correlation or correlation that exists in a large number of data sets, thus describing the laws and patterns that some attributes in a thing appear at the same time[1]. This method has also achieved good application effect in vehicle fault analysis, which can analyze the characteristics of the combination sets of fault phenomena of different vehicle systems and explores the specific correlation of each failure phenomenon based on the historical data resources of vehicle failures, at the same time, can prevent the advance failure according to the location of vehicle failure, phenomenon.

2 Model architecture

Fault association analysis is the process of finding frequent patterns, associations, correlations and causal structures between data items set in the fault data carrier or rather that is the connection between fault data factors. Fault association analysis model according to the fault occurred in each system, which provides an analysis algorithm model for the
correlation between fault location and service time (single factor), the correlation between fault location and fault time, fault environment (double factors), the correlation between fault location and fault time, fault environment, fault mode (three factors). The computational model structure is shown in figure.

![Correlation Analysis Model Diagram](image)

**Figure 1.** Model architecture diagram.

### 3 Algorithm description

#### 3.1 Relevant concepts

Fault association analysis algorithm is to analyze a large number of fault data, from which to find a certain degree of support and credibility of the data between the link rules. The realization of the correlation analysis model in the system is mainly based on Apriori correlation algorithm. Apriori algorithm is an association rule discovery method, which focuses on finding out the occurrence of certain events in the data together to find those trusted and representative rules. In fault analysis, it is mainly used to reveal the correlation of fault occurrence, which can find rules that are credible and representative. The rules are used to reveal the relevance of failures, for example, we found that fault motorization hours ranged from 10 hours to 150 hours, which is prone to oil leakage.

In the process of algorithm realization, it is mainly to find out the relation of item set in the database using the iterative method of layer-by-layer search in order to form strong rules, whose process consists of connections and pruning\(^1\).

**Define:** a set of items is called an itemset. An itemset containing \(k\) items is called a \(k\) items itemset\(^2\).

\{computer, antivirus_software\}: two items itemset

\{computer, antivirus_software, paint_software\}: Three items itemset

The frequency of itemset is to count the number of transactions in the item set, which is called the frequency of itemset, support count, or count. If the relative frequency of occurrence of itemset \(I\) is greater than or equal to minimum support threshold, so it is frequent itemset.

#### 3.2 Apriori algorithm process

The process of this algorithm is actually two major steps: The first step is to discover frequent itemset, and the second step is to produce association rules with confidence greater
than the minimum confidence according to frequent itemset. Here we need to use the concepts of support, minimum support, confidence, minimum confidence, and frequent itemset. Generally, we use supp for support and CONF for confidence. In each instance, minimum support and minimum confidence are specified according to specific industry rules, which are also related to the number of itemset and the size of sample data. The specific steps are as follows:

### 3.2.1 Discover frequent itemset

L is generally used to represent frequent itemset and C to represent candidate itemset\(^2\).

First step, generate frequent itemset \(L_1(k=1)\).

Next step, two frequent sets of \(k\) items \(L_k\) merge to form a frequent set of \(k+1\) items’ candidate sets, namely \(C_{k+1}\).

Next step, pruning \(C_{k+1}\), form frequent itemset of \(L_{k+1}\).

Last step, Stack \(k\) up \(k=k+1\), repeat b and c, until the \(C_{k+1}\) is an empty set or \(k\) is the maximum number of items.

Two steps are repeated in this process: connecting and pruning.

#### 3.2.1.1 Connecting

The candidate \(k\) items itemset are generated by \(C_{k-1}\) and \(C_{k-1}\) connections in order to find \(C_k\), which is called candidate itemset \(L_k\). Set \(C_1\) and \(C_2\) are itemsets in the \(C_{k-1}\), perform connection \(C_{k-1} \cap C_{k-1}\), among them the elements of \(C_{k-1}\), \(C_1\) and \(C_2\) can be connected\(^2\).

#### 3.2.1.2 Pruning.

\(L_k\) members are not always, The frequent \(k\) items itemset is included in the \(L_k\). Scan the database, determine the number of each candidate set in the \(L\), use \(C_{k-1}\) to cut off non-recurrent items in \(L\) to determine \(C\). Apriori algorithm based on a subset of frequent itemset should also be a property of frequent itemset, which uses an iterative method of layer-by-layer search, \(k\) items itemset is used to derive the \(k+1\) items itemset. The specific algorithm flow is as follows:

First traverse the target data set once, recording the number of occurrences of each item or attribute, that is to calculate the degree of support for each project and collect all items whose support is not less than the minimum user support to form candidate itemset \(L_1\); then link all the elements in the \(L_1\) to form candidate 2 items itemset \(C_2\), reversal the database, calculate the support of each candidate 2 items itemset in the \(C_2\) and collect all projects with no less than minimum user support to form frequent 2 items itemset \(L_2\); then join \(L_2\) to form \(C_3\), traverse the data set to get \(L_3\). Repeat above process until there is no candidate itemset.

In the whole process, a large number of candidate itemsets would generates after multiple cycles. At the same time, the verification link needs to be scanned repeatedly, which may produce a large transaction data set\(^3\).

From the above analysis, it can be seen, that Apriori algorithms require multiple scans of potentially large databases. If the frequent itemset contains 20 items, you need to scan the dataset 20 times. Such a large amount of computation is both time consuming and increasing I/O load.

The traversal process is shown below:
3.2.2 Finding association rules

Once we have the frequent itemset, we look for association rules in the frequent itemset, which can also be called a strong association rule. The search for this association rule would filtrate the items layer by layer, which appear in the frequent itemset at last[^3]. The process is as follows:

In frequent itemset, suppose $x'$, $x''$ are subsets of $x$, and $x = x' \cup x''$, then

$$CONF(X \rightarrow Y) = \frac{supp(XY)}{supp(x)} \quad (1)$$

$$CONF(X' \rightarrow Y \cup X'') = \frac{supp(XY)}{supp(x')} \leq CONF(X \rightarrow Y) \quad (2)$$

The above process can be described as follows:

If the rule $X \rightarrow Y$ doesn't meet the confidence threshold, then the rule $X' \rightarrow Y \cup X''$ must be less than confidence, which is what we call priori property.

For $k$ items frequent itemset, judgment begins with an association rule of one item, whose consequent is 1. If it don't satisfy the confidence threshold, then we should delete all rules, which contain this consequent. If it satisfies the confidence threshold, then it would produce the rules whose consequent is 2. Go on judging until the consequent is $k$-1. Eventually we'll get one or more of the rules we need. That's the end of the correlation analysis. But whether these rules are useful or meaningful depends on how they are applied[^2,^3].

3.3 Algorithm optimization

3.3.1 Optimize the number of frequent itemset

Although Apriori algorithm is the most classical algorithm of data mining, it has a large amount of computation in the calculation process and also increases the burden of data interface. Optimization and improvement of the original algorithm can alleviate the problem of time consuming and load in the calculation process. Two optimization strategy can be adopted to compress data transaction set to improve the efficiency of data calculation.

Item $n$ for each transaction that produces a data set $D$, compare $n$ and $k$ before each scan count. If $n<k$, you can ignore the scan transaction, also set $n=0$; if a transaction of data set $D$ and a subset of its items are not counted during candidate counting, set $n=0$. 

[Figure 2. Data traversal process diagram.](image)
3.2.2 Optimization of pruning

Finding the last frequent item is filtered through minimal support. Each frequent itemset is obtained by iterating up, until the last itemset is supported below the preset threshold. In order to minimize the amount of computation, in the formation of each frequent item set, in addition to using the support degree to filter, but also through the priori property of a set to pruning, in order to improve the computational efficiency. When we get to the n items frequent itemset and we combine the n+1 items, you can use this principle to get rid of these combinations, which are not in the n items frequent itemset using the priori property. This will reduce the number of item sets as you stack up.

4 Algorithm application

As a convenient means of transportation, cars are common in our daily life. Because the frequency of automobile use is more frequent, coupled with the influence of hardware wear, environmental impact, human use factors and so on, the frequency of automobile failure is higher. The causes of these failures are complex and may involve one system or several systems. For the same system fault, there are many kinds of fault phenomena, which may be singly or concurrently occurring. The occurrence of these fault phenomena is not accidental, but there is some correlation.

Fault correlation analysis is the process of finding frequent patterns, correlations and causal structures between data item sets in the historical fault data, it can detect the correlation between potential failures and find the correlation between the potential fault phenomena, provide the necessary support for the analysis of the characteristics of the fault law, and play a good auxiliary decision role in fault prediction. The power system of the automobile is the core of the automobile, and its performance directly affects the efficiency and time of automobile. In the process of practical use, its failure rate is also on the high side, and its failure has complex and diverse characteristics.

After statistical analysis of the historical data in the fault database, we found that five engine system failures are common, namely oil leakage, water leakage, stuck, shake and loose. Now, we extract power system fault information from the fault database and analyze whether there are some fixed correlation between these five faults. In the use of association analysis model, the number of selected data sets should not be too large, which will increase the computational burden and time consumption. But it cannot be too little, which will reduce the reliability of the analysis results. In this application, the number of data sets selected is 100 and the analysis factor items are 5 what were mentioned above. Set initial minimum support of 30% and minimum confidence of 60%.

4.1 Discover frequent itemset

In the calculation process, the fault type is first replaced by a specific letter, Here are five categories of factor projects, A is for stuck, B is for shake, C is for oil leakage, D is for water leakage, and E is for loose. In this way, the retrieval of data is more convenient and it is easier to observe and count. Scan and count the initial data set, produces frequent 1 item itemset. As follows:
After forming candidate itemset C3, scan again for comparison and find, that only one item was greater than 50% confidence, namely {B,C,E}. So we can't join up anymore, so we can stop iterating up. Finally, we have the following table of frequent itemset.

**Figure 3.** The calculation process diagram of frequent item sets.

**Figure 4.** The end result of a frequent itemset.

### 4.2 Finding association rules

Find out the strongest association rule after getting the last frequent itemset based on data relevance mining for the Frequent Itemset. According to the above algorithm description, filter by 70% minimum confidence in order to keep or delete these association rules. Specific processes are described below:

#### 4.2.1 starts with consequent B

CONF[(C, E) → B] = \(\frac{0.52}{0.55} = 0.945\) \hspace{1cm} (3)

Antecedent move back, continue to break down.

CONF[C → (B, E)] = \(\frac{0.51}{0.78} = 0.654\) \hspace{1cm} (4)

CONF[E → (B, C)] = \(\frac{0.61}{0.62} = 0.984\) \hspace{1cm} (5)

According to the above calculation results, rule 3and 5 can be retained, rule 4 can be omitted.

#### 4.2.2 Starts with consequent E

CONF[(B, C) → E] = \(\frac{0.52}{0.61} = 0.852\) \hspace{1cm} (6)

Antecedent move back, continue to break down.

CONF[B → (C, E)] = \(\frac{0.55}{0.75} = 0.733\) \hspace{1cm} (7)
According to the above calculation results, rule 6 and 7 can be retained, rule 8 can be omitted.

4.2.3 Starts with consequent C

\[
\text{CONF}[\{B, E\} \rightarrow C] = \frac{0.51}{0.78} = 0.654
\]

(9)

This rule doesn't have to be extrapolated.

### 4.3 Analytic result

The association rules that need to be extracted are \( C, E \rightarrow B \), \( E \rightarrow B, C \), \( B, C \rightarrow E \) and \( B \rightarrow C, E \). Among them should pay attention to most is the first two rules, the correlation degree is strong. The results show, when \( C, E \) faults occur simultaneously, the occurrence rate of \( B \) faults is higher and when \( E \) failure occurs, the \( B, C \) failure rate is slightly higher.

In practical application, it indicates that the oil will leak out after the release bolt of the engine is loosened. After the oil leakage, the amount of lubricating oil will decrease and the lubricating function will be lost, it will make the engine live cylinder, resulting in engine jitter.

### 5 Conclusion

Association analysis is a simple and practical analysis technique of data mining, and it is an effective tool to discover the relevance or correlation in a large number of data sets, which is also applied to the analysis of vehicle fault phenomena. In the process of analysis, the number of frequent item sets was significantly reduced by optimizing the number of frequent item sets and pruning, and the amount of traversal calculation was also greatly reduced\(^4\). This approach is effective in increasing data access efficiency and reducing IO interface load. It can be seen from the analysis results that the algorithm model can analyze the characteristics of the system fault phenomenon combination set of vehicles and find the specific correlation of each fault phenomenon\(^5\). At the same time, it can also prevent the pre-issued fault according to the fault parts and phenomena of the vehicle, which has a certain practical guiding role in the maintenance of the vehicle.

### References