A Parallel FP-Growth Mining Algorithm with Load Balancing Constraints for Traffic Crash Data

Yang Yang (yangphd@buaa.edu.cn)
Beihang University

Na Tian
Beijing Jiaotong University School of Traffic and Transportation

Yunpeng Wang
Beihang University

Zhenzhou Yuan
Beijing Jiaotong University School of Traffic and Transportation

Research Article

Keywords: Traffic crash, Data mining, FP-Growth algorithm, Load balancing

Posted Date: March 1st, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1311180/v1

Abstract

Traffic safety is an important part of the roadway in sustainable development. Freeway traffic crashes typically cause serious casualties and property losses, being a serious threat to public safety. Figuring out the potential correlation between various risk factors and revealing their coupling mechanism are of effective ways to explore and identity freeway crash causes. However, the existing association rule mining algorithms still have some limitations in both efficiency and accuracy. Based on this consideration, using the freeway traffic crash data obtained from WDOT (Washington Department of Transportation), this research constructed a multi-dimensional multi-level system for traffic crash analysis. Considering the load balancing, the FP-Growth algorithm was optimized parallelly based on Hadoop platform, to achieve an efficient and accurate association rules mining calculation for massive amounts of traffic crash data; Then, according to the results of the coupling mechanism among the crash precursors, the causes of freeway traffic crashes were identified and revealed. The results show that the parallel FP-growth algorithm with load balancing constraints has a better operating speed than both conventional FP-growth algorithm and parallel FP-growth algorithm towards processing big data. This improved algorithm makes full use of Hadoop cluster resources and is more suitable for traffic crash large data sets mining while retaining the original advantages of conventional association rule mining algorithm. In addition, the mining association rules model with the improvement of multi-dimensional interaction proposed in this research can catch the occurrence mechanism of freeway traffic crash with serious consequences (lower support degree probably) accurately and efficiently.

1. Introduction

Traffic crash is a serious threat to public health safety and have become the eighth leading cause of human death. The number of deaths caused by traffic crashes continues to increase, with about 1.35 million people dying every year, and the main cause of death for people aged 5 to 29 is traffic crash [1]. Especially on freeway, the vehicles usually run at a fast speed, once the traffic crash occurs, it is easier to cause serious casualties. The coupling mechanism analysis of the factors influencing traffic crash on the freeway, has a certain significance to improve the traffic safety level for freeway.

With the advent of the era of traffic big data, the technology of traffic crash data collection is becoming mature, which provides a strong data support for the analysis of traffic crash coupling mechanism. It was common to use historical traffic crash data to figure out the coupling mechanism of crashes. The quantity and quality of historical data have a crucial influence on the analysis results. Hamed et al. [2] proposed a random parameter Logit model with heterogeneity to study the interaction between traffic crashes and vehicle holding quantity. By studying the causes of freeway traffic crashes, Yu [3] focused on analyzing the causes of secondary rear-end collisions of freeway traffic crashes. Kwayu [4] et al., tested the universality and interaction among the influencing factors of fatal traffic crashes via structured modeling and network topology analysis, and explored the causes of fatal traffic crashes with the data of fatal traffic crashes in Michigan from 2009 to 2018. Sun et al. [5] established a Bayesian spatial model to reveal the potential spatial correlation between segments.

Association rule mining is a fuzzy recognition and decision method based on data mining technology. It is an efficient data mining method, which can be used to identify the combination of key factors related to crash consequences. Mining association rules for freeway traffic crash data can facilitate figuring out the potential association relationship among the influencing factors. Yang [6–8] proposed a multi-dimensional interaction improved Apriori association rules mining algorithm considering directional constraints and index weights: WOMID-Apriori, based on subjective and objective combined weighting model the interval analytic hierarchy Process (AHP) and grey relational degree, the weight of data fields was calculated and optimized, then, the algorithm was applied to address the cause analysis of freeway crashes. Singh [9] et al. improved the performance of Apriori algorithm and proposed two improved Apriori algorithms based on MapReduce: VFPC (Elapsed Time-based Dynamic pass-counting) and ETDPC (Elapsed Time-based Dynamic pass-counting). Wang [10] et al. discussed the algorithm for sorting variables in the data set and proposed two new technologies, MG and SMGR. These two technologies were developed for public safety using classified traffic crash data, which were more intuitive than existing technologies. Yu et al. [11] proposed a new DSTGCN mixed spatio-temporal graph convolutional network to predict traffic crash. Jiang [12] et al. proposed a framework based on association rule mining to identify the factors related to the severity of motorcycle injuries. In order to objectively determine the threshold value of parameters, they developed a parameter optimization method. Huang et al. [13] proposed a parallel FP-growth algorithm based on cloud computing. Samerei [14] et al. worked on the bus crash data set of Victoria State, Australia from 2006 to 2019, divided the traffic crash data into clustering, extracted the factors affecting bus traffic crash fatality based on association rule mining algorithm. Montella [15] et al. used classification trees and association rule mining algorithm to analyze 78,611 crash data involving two-wheeled vehicles in Spain, aiming to extract knowledge from a large amount of data and identify understandable association rules. Zheng et al. [16] applied the gradient lifting algorithm to analyze the relationship between collision severity and influencing factors. Bechini et al. [17] proposed a distributed association rule mining algorithm based on MapReduce programming model, confirmed the scalability and performance improvement of this method on small computer clusters.

Global scholars have made some achievements in the research of traffic crash analysis methods and technologies so far. However, there are also some deficiencies in these studies: (1) Most of the algorithms currently applied in the field of traffic crash data mining are ordinary serial algorithms, a huge number of candidate item sets may be generated in the mining process. The support of the algorithm during the calculation requires a large amount of memory, and it is unable to efficiently mine the massive traffic crash data. In addition, the ordinary parallel association rule mining algorithm generally has the problem of load imbalance, which may lead a low computing efficiency. Most of the existing research improved the algorithm by adding constraint conditions to achieve the efficient extraction of mining results, but lack of improvement on the performance of mining algorithm itself. (2) Existing association rules mining basically focused on the high degree level of support and confidence of association rules, while only few studies focused on the combination rules with low support and high confidence those were prone to lead to serious consequences such as casualties, which may lead that the potential important traffic crash

precursors missed. However, the characteristics and mechanism of serious casualty crashes and ordinary crashes may be different, so the conventional safety improvement measurements for ordinary crashes may not work for the serious crashes.

In view of the above defects, based on Hadoop platform, this research proposed a parallel FP-growth algorithm considering load balancing for freeway traffic crash data, which is used to mine multi-dimensional and multi-level massive crash data. Then the association rules with high support and confidence from mining results were extracted and interpreted. Finally, based on the mining results, the suggestions to improve freeway traffic safety were put forward.

2. Data Process

2.1 Data source description

The data source of this research is the UW DriveNet Intelligent Transportation Big Data platform of the University of Washington in the United States. Through the connection between the platform and WDOT, a total of 345,545 pieces of freeway traffic crash data in Washington state can be obtained, including 384 attributes, specifically includes: Driver's gender, age, driver's degree of soberness, crash location, vehicle type, type of airbag, state of airbag ejection, weather, road type, traffic control, crash time and season, etc.

2.2 The establishment of multi-dimensional and multi-layer data system

The multi-dimensional and multi-layer traffic crash coupling mechanism analysis system is the basic framework of association rule mining. In this research, crash influencing factors are taken as the first layer of the system, including six dimensions "driver, vehicle, road, environment, time, crash". In the six dimensions of the first layer of source data, specific influencing factors with analytical value are screened out respectively. As the second layer, there are 29 dimensions in total. The attributes of the influencing factors in the second layer are further refined, and the specific indexes and values of the attributes are clarified, forming the third layer of the system. The specific indexes and values of the crash dimension are shown in Table 1.

Table 1
Specific indexes and values of influencing factors of collision dimension

Factors	Values	Codes of Valus
Injury Type	No Injury	AH1
	Possible Injury	AH2
	Serious Injury	AH3
	Death	AH4
Ejection Status	Not Ejected	W1
	Unknown if Ejected	W2
	Partially Ejected	W3
	Totally Ejected	W4
Collision Severity	PDO	AE1
	Injury	AE2
	Fatality	AE3
Towed Indicator	No	D1
	Yes	D2
Collision Report Type	Suburban	A1
	Urban	A2
	Interstate	A3
	Ramp	A4

Based on the dimensional and hierarchical analysis results of the above crash attributes, the construction of multi-dimensional OLAP database and data mining in THE SQL Server Analysis Service (SSAS) of the database management system should be addressed. The specific steps are as follows:

(1) Import data preprocessed with Excel into Access.

(2) The fact table (freeway traffic crash) and 6-dimension tables (driver, vehicle, road, environment, time, crash) are constructed, the primary keys of each table are demarcated, and the relationship between the out keys of fact table and the primary keys of each dimension table are set up. Finally, the Access database is promoted to SQL Server database.

(3) Create the research project "Freeway Traffic Crash" in SSAS, import the data source.

(4) Set the ID attribute of the fact table "freeway traffic crash" as the metric value and "Count" as the aggregation mode, built the multi-dimensional database and dimension considering the data analysis requirements, and deploy the project to the server.

3. Design And Implementation Of The Algorithm

The FP-Growth algorithm constructs the data set as FP-tree and stores it in memory, then mines frequent item sets by recursively calls condition of FP-tree [18]. However, when mining massive data with low support degree, the FP-tree generated by FP-growth will occupy a large amount of memory, causing problems such as memory overflow and long operation time [19–20]. This research tries to solve the above problems through the optimization design of the parallel FP-growth algorithm, proposes a load balancing optimization scheme based on the parallel FP-growth algorithm, and builds an algorithm experimental environment under condition of the Hadoop framework. Through experiments, the efficiency of traditional FP-growth algorithm, parallel FP-growth algorithm and parallel FP-growth algorithm considering load balancing with different data volume and different minimum support threshold are compared and analyzed respectively.

3.1 Principle of parallel FP-Growth algorithm

The performance bottleneck of FP-growth algorithm when processing massive data is solved using horizontal partitioning [19]. The horizontal partitioning method is to divide the database into various sub-database nodes, make frequency statistics for each sub-node, build local FP-tree, obtain the conditional pattern base of each data frequent item set, gather various conditional pattern bases to the corresponding nodes, and then build conditional FP-tree for each node, operate the local FP-growth mining. The local FP-tree diagram considering the influencing factors of traffic crashes is shown in Fig. 1.

3.2 Process design of parallel FP-growth algorithm

The operation process of the parallel FP-growth algorithm can be summarized into the following five steps, and the algorithm process is introduced in combination with this research case. The specific process design is shown in Fig. 2:

Step 1, Shard: The freeway historical crash database of Washington state is horizontally divided into six consecutive sub-databases from the dimensions of driver, vehicle, road, environment, time and crash, which are stored on several different computers respectively. Each sub-database obtained after segmentation is defined as slice.

Step 2, Parallel counting: Read each slice obtained in Step 1 and use "Mapreduce. job" to count frequent "1- itemset FList". Where the "Mapper" function starts to input with the key pairs of < The key, value = T_i>, after processing, transaction T_i is output as key pairs < Key = a_j, value = 1>, where AJ is the J_{th} data item of T_i; All slices are processed by Mapper function, and the key pairs with the same key value are summed

by Reduce function. The key values output by the Reduce function are sorted in order of frequency from high to low. Finally, FList is obtained according to the minimum support filtering.

Step 3, Grouping for Frequent 1- item sets: All frequent items in FList are evenly divided into GROUP G, in which each group has a unique group ID- Gid. Frequent items and their corresponding group numbers constitute frequent item table group GList and are assigned into all slices.

Step 4, design parallel FP-growth: This step is the core step of the parallel algorithm. Partial FP-growth mining is operated through a MapReduce. Job, where the Mapper function is defined by key pairs of < The key, value = T_j> to read each slice. Scan all frequent items A_j in transaction T_j, find the corresponding group ID identifier Gid in the corresponding frequent item table group GList, and output it with the key value. The output of Mapper function is read through the Reduce function corresponding to different Key values, local FP-tree is built, and local FP-growth mining is carried out for frequent items corresponding to each Gid in a recursive way. The frequent item sets obtained from mining are saved in a heap in order from large to small, and the top K frequent item sets with the highest support are finally returned.

Step 5. Aggregation: Read the output of Step 4, combine the same key-value pairs via the Mapper function, and then use the Reduce function to calculate and output the frequent item sets of the top K items with the highest global support and confidence.

3.3 Implementation of optimized parallel FP-growth algorithm with load balancing constraints

3.3.1 The load balancing definition and grouping strategy

Load balancing refers to the balancing of loads (work tasks) and spreading them across multiple units of operation so that work tasks can be completed cooperatively. The load balancing defined in this research refers to the balanced grouping of frequent 1-itemsets to achieve the relative balance of the running load for each slice, so as to improve the efficiency of data mining. The core strategy of load balancing mainly includes two steps: The first step is to build a load model and recursively mine the total workload of FP-tree on the projection database for each frequent data item. The second step is to divide all frequent items into G groups by balanced grouping strategy and divide transaction T into G groups to ensure the load balancing of each node. Tasks are completed by corresponding nodes of various groups, and FP-tree is recursively mined for frequent items, so as to improve the processing capability of the system.

(1) Establishment of load model

For the FP-growth algorithm, it is not realistic to accurately calculate the workload of each frequent item, we can only reasonably estimate the workload. The workload of recursively mining the condition FP-tree of each frequent item is equal to the times that the condition FP-tree of the frequent item recursively calls FP-growth. The length of the longest frequent path of each frequent item in the conditional FP-tree is equivalent to the location of the frequent item in the FList.

In the conditional FP-tree, the number of times that the same frequent item recurrently calls FP-growth has an exponential relationship with the length of its longest frequent path [18–20], and the load model formula (1) can be obtained, where F_i is the load estimated value of each node, and P (I, FList) represents the position of the frequent item in FList of frequent 1-itemset:

$F_i = \log P(_{i, FList})$ (1)

(2) Balanced grouping

According to formula (1), in frequent 1-itemset FList, the higher the support count of frequent items is, the higher its position is, and the larger its load value is. Therefore, the load value of frequent items in FList forms an increasing trend. Since it is difficult to achieve a globally optimal grouping, this research uses a greedy strategy to group frequent items. Assuming that the quantity of item groups is G, FList is grouped in order from the back to the front. First, the first G frequent items are put into G groups, and then the frequent items to be grouped are put into the group with the smallest total load in sequence. At the same time, the load value of the frequent item is accumulated to the total load of the group. Repeat the steps until all the frequent items in FList are allocated.

3.3.2 Implementation of load balancing algorithm

The process of the improved load-balancing parallel FP-growth algorithm is the same as that of the above parallel FP-growth algorithm. Only the frequent 1-itemset grouping in the third step is improved. The pseudo-code and flow chart of the algorithm are shown in Table 2 and Fig. 3 respectively.

Table 2
Code implementation of load balancing algorithm

Load balancing algorithm pseudocode:
generateGList Function:
1. void generateGList(List FList, int G){
2. HashTable GList;
3. Heap minHP;
4. groupsNum ← G;
5. Gid ← 0;
6. if FList.length() < groupsNum then
7. foreach item a_j in FList do
8. GList.put(a_j, Gid);
9. Gid ← Gid + 1;
10. end
11. else
12. for j = 0 to groupsNum do
13. F(item[j]); // Load estimation
14. minHP.add(item[j]);
15. minHP.adjust();
16. end
17. for j = groupsNum to FList.length() do
18. F(item[j]); // Load estimation
19. minHP[0].weight ← minHP[0].weight + item[j].weight;
20. GList.put(item[j], minHP[0].group);
21. minHP.adjust();
22. end
23. end
24. }

3.4 Construction of experimental environment

In this research, there is a large amount of historical crash data on freeways, so the complete distribution mode is selected in the experiment to build a Hadoop distributed processing system. A cluster is built with three computers installed with Linux system. The cluster has three nodes in total, including one master node and two slave nodes. The hardware and software configuration of each computer is shown in Table 3.

Table 3
Virtual machine configuration

The hardware configuration	The software configuration
CPU: Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.2GHz	Linux: Ubantu 18.04.5
Memory: 32GB	Java version: JDK-8U161-linux-x64
Hard Disk: 360GB	Hadoop version: Hadoop-2.7.4
GPU: AMD_Radeon R7 350	Integrated development environment: Eclipse

The process of building a Hadoop framework cluster is as follows:

Step 1: Install Linux Ubantu 18.04.5 on the three computers respectively, change the host name and network configuration to "master", "slave1", and "slave2".

Step 2: Modify the hosts file on the computer, set the public and private keys to implement SSH login without password between computers.

Step 3: Upload and install JDK and configure environment variables for these computers.

Step 4: Upload and install Hadoop, perform operations on environment variables and configuration files in the conf file of the computer, and format HDFS.

Step 5: Start the Hadoop cluster and run the JPS command to view all processes. The Hadoop cluster process starts as shown in Fig. 4.

3.5 Comparison of improved algorithm efficiency

In order to verify the rationality of the algorithm designed in this research, the serial FP-growth algorithm, parallel FP-growth algorithm, the parallel FP-growth algorithm with load balancing constraints are respectively run under the Hadoop cluster. In this experiment, the computational efficiency of the algorithm is compared and analyzed from the perspectives of data volume and minimum support.

(1) Sensitivity analysis for data volume on algorithm efficiency

Various records are randomly selected from the data set to constitute the data set, and the four virtual processed data sets used in the experiment include 100,000, 500,000, 1,000,000 and 1,500,000 records respectively. The results of running the three algorithms with a minimum support of 0.5 are shown in Fig. 5.

Figure 5 indicates that, when the experimental data set is small, the running time difference of the three algorithms is very small. The possible explanation is that the two parallel FP-growth algorithms need extra time to start the cluster, which fails to take advantage of the parallel algorithm. With the increase of the data set, the operation time of the two parallel FP-growth algorithms is greatly reduced compared with the serial algorithm, and the performance advantage of the algorithms is more obvious with a larger number of records. Compared with the parallel FP-growth algorithm without load balancing, the operation time of the parallel FP-growth algorithm with load balancing based on balanced grouping method can be further reduced, which is more suitable for mining large data sets. The experimental results in Fig. 5 can prove that the optimized load-balancing parallel FP-growth algorithm can not only solve the problems such as large memory consumption and long running time when the traditional serial algorithm performs big data mining, but also further shorten the data mining time and improve the efficiency of the algorithm.

(2) Sensitivity analysis for the minimum on algorithm efficiency

According to the previous analysis of association rules and algorithms in this research, the setting of minimum support degree has a great impact on the operation efficiency of the algorithm. The smaller the minimum support, the greater the number of frequent items, and the longer the operation time of the algorithm. In this experiment, the data set with 500,000 records in experiment (1) is selected, and three algorithms are adopted for data mining under the condition that the minimum support is set at 0.1–0.7. The calculation time comparison is shown in Fig. 6.

Figure 6 indicates that, when the minimum support is high, such as 0.6 and 0.7, the number of frequent items in the data set is small, and the operation time of the three algorithms is short. With the decrease of support, the number of frequent items in the data set increases, and the operation time of the algorithms begins to rise. When the minimum support is lower than a certain degree, the serial FP-growth algorithm has the problem of memory heap overflow. At this time, the algorithm will automatically report an error and stop running. However, the parallel FP-growth algorithm overcomes this shortcoming, and can still run under a lower support threshold, its running time is less than the serial algorithm. According to the overall running time of the algorithm, the running efficiency of the optimized parallel FP-growth algorithm is higher than that of the original one, its advantages become more prominent as the support threshold decreases.

The above experimental results show that the optimized parallel FP-growth algorithm based on Hadoop proposed in this research can perform efficient data mining in the case of low support. The operating efficiency of the improved algorithm is obviously better than that of the serial algorithm and the original parallel FP-growth algorithm, and it can be well applied to the subsequent data mining for freeway traffic crashes in this research.

4. Cause Analysis Of Freeway Traffic Crashes Based On Improved Fp-growth Algorithm

During the process of FP- Growth association rule mining algorithm, parameter threshold Settings is closely related to the efficiency and quality of mining, considering the coupling mechanism between precursors related to serious casualty as high practical significance with the traffic safety, consequence with the values of "death or serious injury" in the original data set actually has a lower percentage actually. If the minimum support threshold is set too high, death or serious injury both attribute variables may be cut off in the pruning process. Therefore, in this study, the minimum support threshold will be set based on the occurrence frequency of the casualty attribute and its related attributes, as far as possible to efficiently and accurately mining association rules effectively.

4.1 Mining Results and analysis

Multi-dimensional interaction association rule mining refers to that the directed association mining in the form of "LHS = > RHS" is the association rule of constraint condition. After a lot of tests, set the minimum support threshold as 0.05 (in case ignoring the traffic crashes with severe consequences but lower frequency), minimum confidence threshold is set as 0.4, the lift degree threshold to set as 1. The variables in the dimension of "Driver, vehicle, road, environment, time" are constrained in LHS, the crash dimension variables are constrained in the RHS. Then the association rules with high support and high confidence obtained from mining are extracted respectively, as shown in Tables 4 and 5.

Tab.4 Mining results of association rules (In descending order of support)

Rules (LHS=>RHS)	Sup	Conf	Lift
{Speeding=N1,Drowsy=Q1,Vehicle Condition=AG1, Unlicensed=P1,Distracted=O1}{Number of Motor Vehicles Involved=AD2,Ejection Status=W1}	0.529	0.901	1.013
{Speeding=N1,Age=J3,Drowsy=Q1,Vehicle Condition=AG1, Unlicensed=P1,Distracted=O1}{Number of Motor Vehicles Involved=AD2}	0.522	0.919	1.074
{Speeding=N1,Vehicle Condition=AG1,Roadway Surface Condition=I1,Weather=Z1,LightingCondition=AA3}{Ejection Status=W1,Injury Type=AH1}	0.507	0.782	1.016
{Gender=K2,Speeding=N1,Drowsy=Q1,Roadway Surface Condition=I1} {Ejection Status=W1,Injury Type=AH1}	0.489	0.891	1.006
{Speeding=N1,Restraining System Type=X1,Lighting Condition =AA3}{Ejection Status=W1,Number of Motor Vehicles Involved=AD2}	0.449	0.827	1.127
{Speeding=N1,Drowsy=Q1,VehicleCondition=AG1,Roadway Surface Condition=I1}{Ejection Status=W1,Injury Type=AH1}	0.377	0.907	1.023
{Age=J2,Speeding=N1,Drowsy=Q1,Vehicle Condition=AG1, Sobriety Level=L1,Weather=Z1}{Injury Type=AH1}	0.368	0.877	1.077
{Gender=K1,Speeding=N1,Drowsy=Q1,Vehicle Condition =AG1,Unlicensed Driver=P1,Roadway Surface Condition} {Ejection Status=W1,Injury Type=AH1}	0.329	0.659	1.109
{Speeding=N1,Drowsy=Q1,Vehicle Condition=AG1,Sobriety Level=L1} {Ejection Status=W1,Injury Type=AH1}	0.322	0.792	1.074
{Quarter Number=G1,Speeding=N1,Vehicle Condition=AG1, Lighting Condition=AA3}{Towed Indicator=D1}	0.317	0.843	1.008
{Quarter Number=G3,Vehicle Condition=AG1,Unlicensed Driver=P1,Lighting Condition=AA3}{Injury Type=AH1}	0.310	0.891	1.015
{Speeding=N1,Drowsy=Q1,Roadway Characteristic=B1, Vehicle Action=AF2}{Collision Severity=AE1}	0.303	0.922	1.018
{Speeding=N1,Drowsy=Q1,Vehicle Condition=AG1,Unlicensed Driver=P1} {Number of Motor Vehicles Involved=AD2}	0.293	0.864	1.152

Through the analysis of the results in Table 4, the following conclusions can be drawn: Traffic crashes in the study area are mostly caused by passenger cars, most of which involves male drivers. Minor crashes those do not cause casualties account for the majority, and most of them occur on urban or interstate freeways. Most of the crashes involve only 1–2 vehicles. Traffic crashes those do not result in injuries are

probably not caused by drivers' driving violations. A possible explanation for the higher incidence of minor crashes those do not result in injuries is that bad weather is less common in summer and roads' conditions are basically dry. Non-fatal crashes usually do not cause airbags ejection. Most of the traffic crashes those occur when the vehicle is going straight may result in minor traffic crashes with no casualties.

Tab.5 Mining results of association rules under (In descending order of confidence)

Rules (LHS=>RHS)	Conf	Sup	Lift
{Age=J2,Weather=Z2,Roadway Surface Condition=I2} {Collision Severity=AE2}	0.968	0.158	27.970
{Day Of Week=F1,Quarter Number=G1,Hour=H1,Roadway Surface Condition=I3}{Number of Motor Vehicles Involved=AD3}	0.961	0.057	37.801
{Age=J1,Speeding =N2,Unlicensed=P2,Lighting Condition=AA2} {Collision Severity=AE3}	0.954	0.071	86.598
{Gender=K2,Sobriety Level=L3,Roadway Surface Condition=I7} {Collision Severity=AE3}	0.939	0.069	78.723
{Age=J2,Day Of Week=F1,Hour=H1,Roadway Surface Condition =I3}{Collision Severity=AE2}	0.937	0.125	18.652
{Vehicle Type=T2,Vehicle Condition=AG2,Distracted=O1,Speeding =N2} {Towed Indicator=D2,Collision Severity=AE3}	0.930	0.271	3.672
{Day Of Week=F2,Quarter Number=G4,Hour=H2}{Injury Type =AH1}	0.927	0.193	1.162
{Speeding=N1,Drowsy=Q1,Roadway Characteristic=B1,Vehicle Action=AF2}{Number of Pedestrians Involved=AC1,Collision Severity=AE1}	0.922	0.303	1.018
{Speeding=N1,Age=J3,Drowsy=Q1,Vehicle Condition=AG1, Unlicensed=P1,Distracted=O1}{Number of Motor Vehicles Involved=AD2}	0.919	0.522	1.074
{Age=J4,Drowsy=Q2,Vehicle Type=T1}{Number of Motor Vehicles Involved=AD2}	0.903	0.081	63.190

The following conclusions can be drawn from the results in Table 5: When the driver has dangerous driving behaviors such as overspeed driving, fatigue driving, distracted driving, unlicensed driving or drunk driving, and the road surface is wet or snowy, the traffic crashes easily tend to cause casualties. When the driver's dangerous driving behavior and slope road appear at the same time, it is also very easy to cause serious consequences of casualties. The combination of unsafe driving and bad weather often leads to serious injury or death. Vehicle failure and bad weather at the same time tend to cause

casualties. Among the traffic crash, drivers under the age of 25 are often associated with speeding, and drivers over the age of 65 are often associated with distracted driving. Most of the association rules extracted above include driver factors, so it indicates that human is the main operator in traffic and the key factor that determines whether a crash occurs or not.

4.2 Suggestions for improving freeway traffic safety based on association rule mining

The occurrence frequency of association rules with high confidence is typically less than that of association rules with high support. The LHS occurrence of this kind of association rules can easily cause the occurrence of the RHS, but if any field variables in LHS changes, the association rule may no longer hold. Therefore, the control principle for these kind of association rules is to control from LHS and RHS simultaneously. The following suggestions are made for low-support high-confidence association rules that cause serious injuries:

(1) Rain or snow weather + large van + curve with slopes

Rain and snow may interfere with the driver's vision and visibility, and on the one hand, it may be accompanied by ice or wet on the road, both of which will lead to an increase in the braking distance of the vehicle. Large trucks are relatively heavy in their own inertia, and in the rain and snow, they usually have the risk of rushing out of the road or crashing into nearby vehicles when they are in a crash-prone situation, both of which have serious consequences. In this case, we can avoid the occurrence of RHS by controlling the occurrence of LHS: the first is to optimize the design of the mark line in the curve with slope, set up the speeding capture equipment, and control the speed on the dangerous road. Secondly, we should increase the inspection for large truck overload, and set up the emergency safe lane on the dangerous road such as the continuous slope, and the surface of the road should be made of small stones, so that the vehicles in the emergency can slow or stop immediately, thereby avoiding the occurrence of the crash or reducing the severity. In addition, the safety guard should be installed to prevent the vehicle from sliding out of the road when it collided in a detour, the monitoring for road environment should be improved, severe weather that threatens road traffic safety should be forecasted timely, road cleaning and maintenance during rain and snow should be should be taken into account, such as by sprinkling industrial salt to melt and clear the snow on the road.

(2) Tire failure + passenger car + peak hours

Traffic crashes of passenger cars with tire faults during peak hours usually lead to injuries and most of them involve multiple motor vehicles. On the one hand, the traffic volume during peak hours is large and the driving distance between vehicles is small. On the other hand, the passenger cars with tire faults may be uncontrolled and easily collide with surrounding vehicles.After the collision, the surrounding vehicles are prone to serial rear-end collision because of a small distance between vehicles, consequently resulting in injuries. For this kind of situation, related department should guide the vehicle owner regularly check, increase the frequency of the motor vehicle compulsory inspection or strictly control the test process,

minimize the vehicle potential safety hazard as much as possible. In addition, the traffic volume during peak hour is large, which is prone to congestion, policies such as encouraging off-peak commuting can be adopted to realize the time separation of traffic flow.

(3) Main line + straight road with slope + dust or sand blowing weather

The majority of crashes on the Main line result in injuries due to dust or sand-blowing weather, which can affect drivers' vision and sight. Druing the periods of design, construction and management traffic safety should be the first consideration. For the continuous steep slope, steep curve, long down ramp and long straight road and other dangerous roads, traffic safety risks should be eliminated timely. Early warning should be given to bad weather such as dust or sand blowing, and monitoring of road environment should be strengthened.

(4) Drivers over 65 + distracted driving + flat road

Most of the 65 + aged drivers have some impairment of vision and hearing, most crashes involving distracted driving on straight roads involve death or injury. For this situation, elderly drivers should be subjected to an annual physical examination, and their driving licenses should be revoked if they do not meet the conditions. For drivers with distracted driving, some stimuli can be used to make them keep their attention in high concentration, color road signs may work to stimulate their vision, speed bumps can be set to stimulate their senses, and warning lights or slogans can be installed to attract their attention. In road design, long straight lines should be avoided. If unavoidable, raised warning signs must be set up on the road surface every certain distance to prevent the driver from appearing shallow sleep due to the monotonous driving process.

The mining results with high support are mostly minor traffic crashes without casualties, indicating simultaneous occurrences of a certain attribute combination account for a high proportion in the total number of items. The safety control principle for such crashes is to avoid RHS by controlling LHS.

5. Conclusions

This research proposes a parallel FP-growth association rule mining algorithm considering load balancing, compares the operational efficiency with serial FP-growth algorithm and conventional parallel FP-growth algorithm using various data set volumes and support threshold degree respectively. Based on the proposed association rule mining algorithm considering load balancing, the freeway traffic crash data applied in this research is analyzed with the consideration of directional constraints under multi-dimension and multi-level perspective, the results of association rule mining with high support and high confidence (including low support) are obtained. The specific conclusions of this research are as follows:

(1) Compared with serial FP-growth algorithm or traditional parallel FP-growth algorithm, the proposed parallel FP-growth association rule mining algorithm considering load balancing has more advantages in computational efficiency, and it can reduce the possibility of leading to memory heap overflow when

processing massive data or low support threshold setting work. It is more suitable for mining large traffic crash data sets.

(2) Serious consequences data pieces in crash data sets may be assigned a low degree of support (which may have a high confidence degree) when the association rule mining algorithm is applied, these crashes are easy to be ignored. The algorithm proposed in this research has great advantages in mining association rules with low support threshold. It can overcome the shortcomings of the mining work via traditional association rule mining algorithm towards the data set including serious traffic crash consequence, as well as better reveal the coupling mechanism among the factors related serious traffic crash.

(3) Association rules with high confidence indicate that the association rules have high conditional probability: for the situation combinations with high confidence appear in the LHS, attention should be paid, so as to prevent the serious consequences of traffic crashes. An association rule with high support indicates the high occurrence frequency of the association rule. For an association rule with high occurrence frequency but minor crash consequence, the LHS should be controlled to prevent the consequences in RHS.

(4) Traffic safety is an important element of sustainable transportation, and association rule mining algorithm is an effective method in the field of fuzzy decision. The approach proposed in this research can provide theoretical guidance for fuzzy identification of sustainable transportation.

Due to the limitation of experimental conditions, this research adopts three computers to build a fully distributed experimental mode of Hadoop. This method can realize the database association rule mining, but it is not enough to analyze the impact of cluster node number on operation efficiency. In the future research, the experimental environment should be improved to get more accurate experimental data.

Declarations

Compliance with Ethical standards:

Humans and animals are not involved in this research work.

We used our own data.

Funding

This work was supported by China Postdoctoral Science Foundation [Grant NO. 2021M700333].

Competing Interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. World Health Organization (2018) Global status report on road safety 2018: Summary[R]. World Health Organization
2. Hamed MM, Al-Eideh BM (2020) An exploratory analysis of traffic accidents and vehicle ownership decisions using a random parameters logit model with heterogeneity in means[J]. Analytic methods in accident research 25:100116
3. Yu Q (2013) Causes and prevention measures of secondary rear-end accidents in the rescue of highway traffic accidents[J]. Procedia Eng 52:571–577
4. Kwayu KM, Kwigizile V, Lee K et al (2020) Discovering latent themes in traffic fatal crash narratives using text mining analytics and network topology[J], vol 150. Accident Analysis & Prevention, p 105899
5. Sun J, Li T, Li F et al (2016) Analysis of safety factors for urban expressways considering the effect of congestion in Shanghai, China[J], vol 95. Accident Analysis & Prevention, pp 503–511
6. Yang Y (2020) Research on the Method of Freeway Crash Risk Identification and Comprehensive Traffic Safety Evaluation Considering the Regional Type Difference[D]. Beijing Jiaotong University
7. Yang Y, Yuan ZZ, Sun DY et al (2019) Analysis of the factors influencing highway crash risk in different regional types based on improved Apriori algorithm. J Adv Transp Stud 49:165–178
8. Yang Y, Yuan Z, Chen J et al (2017) Assessment of osculating value method based on entropy weight to transportation energy conservation and emission reduction. J Environ Eng Manage J 16:2413–2424
9. Singh S, Garg R, Mishra PK (2018) Performance Optimization of MapReduce-based Apriori Algorithm on Hadoop Cluster[J].Computers & Electrical Engineering, :348–364
10. Thangaraj DVSJJ, Khanna MR "An Improved Early Detection Method of Autism Spectrum Anarchy using Euclidean Method," 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2020, pp. 1173–1178, doi: 10.1109/I-SMAC49090.2020.9243361
11. Yu L, Du B, Hu X et al (2021) Deep spatio-temporal graph convolutional network for traffic accident prediction[J]. Neurocomputing 423:135–147
12. Jiang F, Yuen KKR, Lee EWM (2020) Analysis of motorcycle accidents using association rule mining-based framework with parameter optimization and GIS technology[J]. J Saf Res 75:292–309
13. Huang Y, Huang J, Liu C et al (2020) PFPMine: A parallel approach for discovering interacting data entities in data-intensive cloud workflows[J]. Future Generation Computer Systems 113:474–487
14. Samerei SA, Aghabayk K, Mohammadi A et al (2021) Data mining approach to model bus crash severity in Australia[J]. J Saf Res 76:73–82
15. Montella A, de Oña R, Mauriello F et al (2020) A data mining approach to investigate patterns of powered two-wheeler crashes in Spain[J], vol 134. Accident Analysis & Prevention, p 105251
16. Zheng Z, Lu P, Lantz B (2018) Commercial truck crash injury severity analysis using gradient boosting data mining model[J]. J Saf Res 65:115–124

17. Bechini A (2016) A Map Reduce solution for associative classification of big data[J]. Inform Sciences: Int J 332:33–55
18. Iko Pramudiono and Masaru Kitsuregawa.Parallel FP-Growth on PC cluster.In PAKDD,2003
19. Le Zhou,Zhiyong Zhong,Jin Chang,Junjie Li,Joshua Zhexue Huang,Shengzhong Feng.Balanced parallel FP-Growth with MapReduce.Information Computing and Telecommunications (YC-ICT),2010 IEEE Youth Conference on 28–30 : 243–246
20. He K, Yuan Z, Yang Y (2022) “A Roadway Safety Sustainable Approach: Modeling for Real-Time Traffic Crash with Limited Data and Its Reliability Verification,” Advances in Transportation Studies, Journal of Advanced Transportation, vol. Article ID 1570521, 2022

Figures

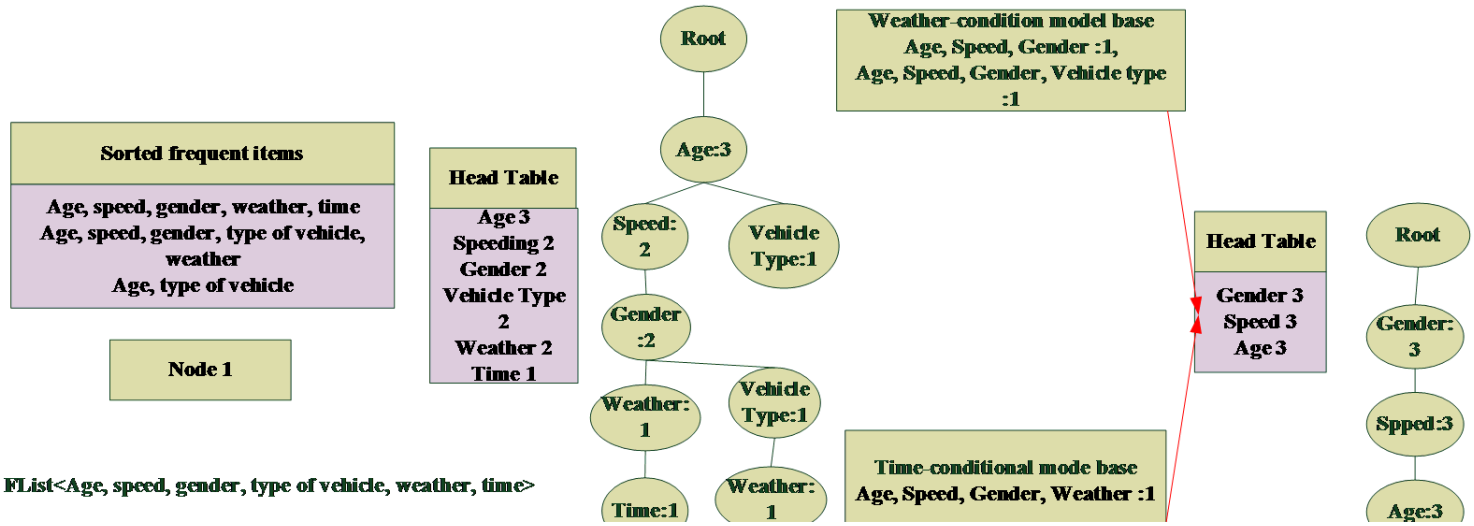

Figure 1

Schematic diagram of local FP Tree in horizontal division mode

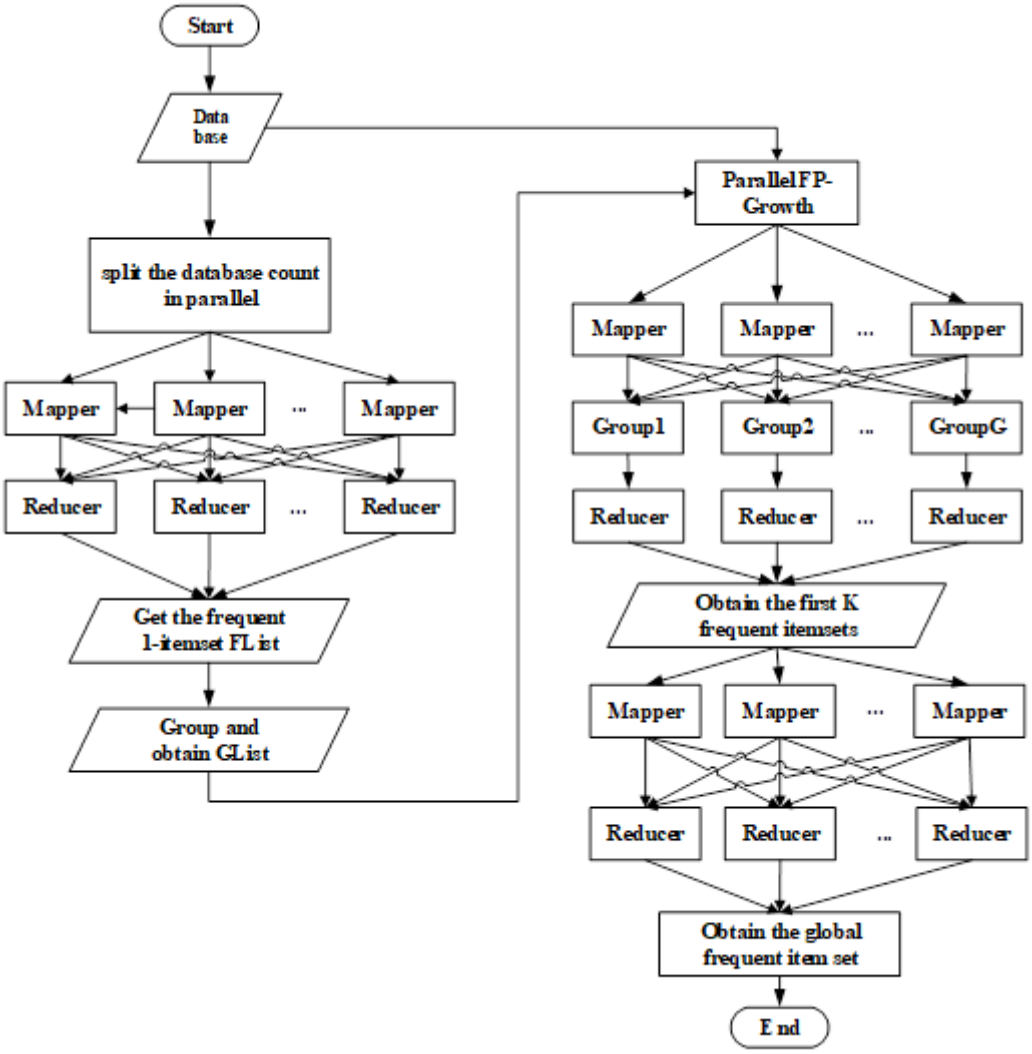

Figure 2

Parallel FP-Growth algorithm flow chart

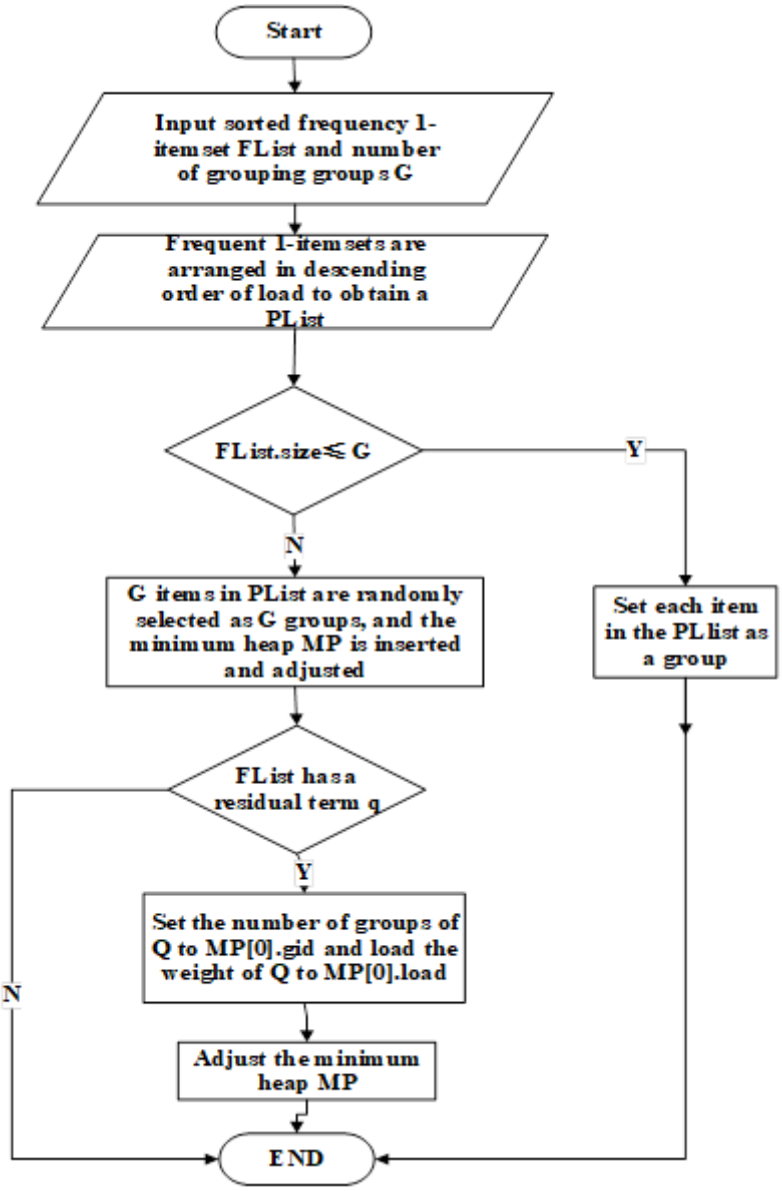

Figure 3

Load balancing algorithm flow chart

```
[root@Master hadoop-2.7.4]# ./sbin/start-all.sh
This script is Deprecated. Instead use start-dfs.sh and start-yarn.sh
Starting namenodes on [Master]
Master: starting namenode, logging to /bigdata/hadoop-2.7.4/logs/hadoop-root-namenode-Master.out
Slave2: starting datanode, logging to /bigdata/hadoop-2.7.4/logs/hadoop-root-datanode-Slave2.out
Slave1: starting datanode, logging to /bigdata/hadoop-2.7.4/logs/hadoop-root-datanode-Slave1.out
Starting secondary namenodes [Master]
Master: starting secondarynamenode, logging to /bigdata/hadoop-2.7.4/logs/hadoop-root-secondarynamenode-Master.out
starting yarn daemons
starting resourcemanager, logging to /bigdata/hadoop-2.7.4/logs/yarn-root-resourcemanager-Master.out
Slave2: starting nodemanager, logging to /bigdata/hadoop-2.7.4/logs/yarn-root-nodemanager-Slave2.out
Slave1: starting nodemanager, logging to /bigdata/hadoop-2.7.4/logs/yarn-root-nodemanager-Slave1.out
[root@Master hadoop-2.7.4]# jps
14048 SecondaryNameNode
14307 ResourceManager
14451 NodeManager
14843 Jps
13630 NameNode
13806 DataNode
```

Figure 4

Hadoop cluster startup results

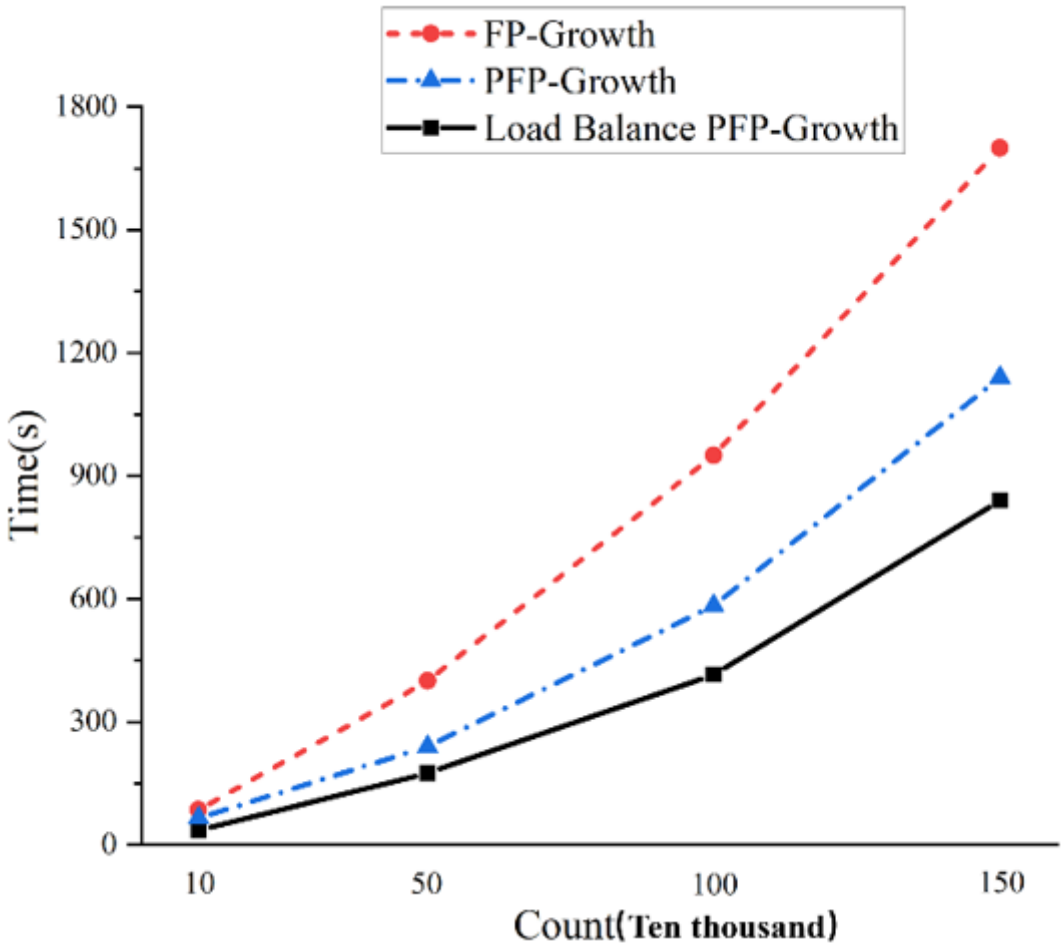

Figure 5

Comparison results of operation time of three algorithms with various data volume

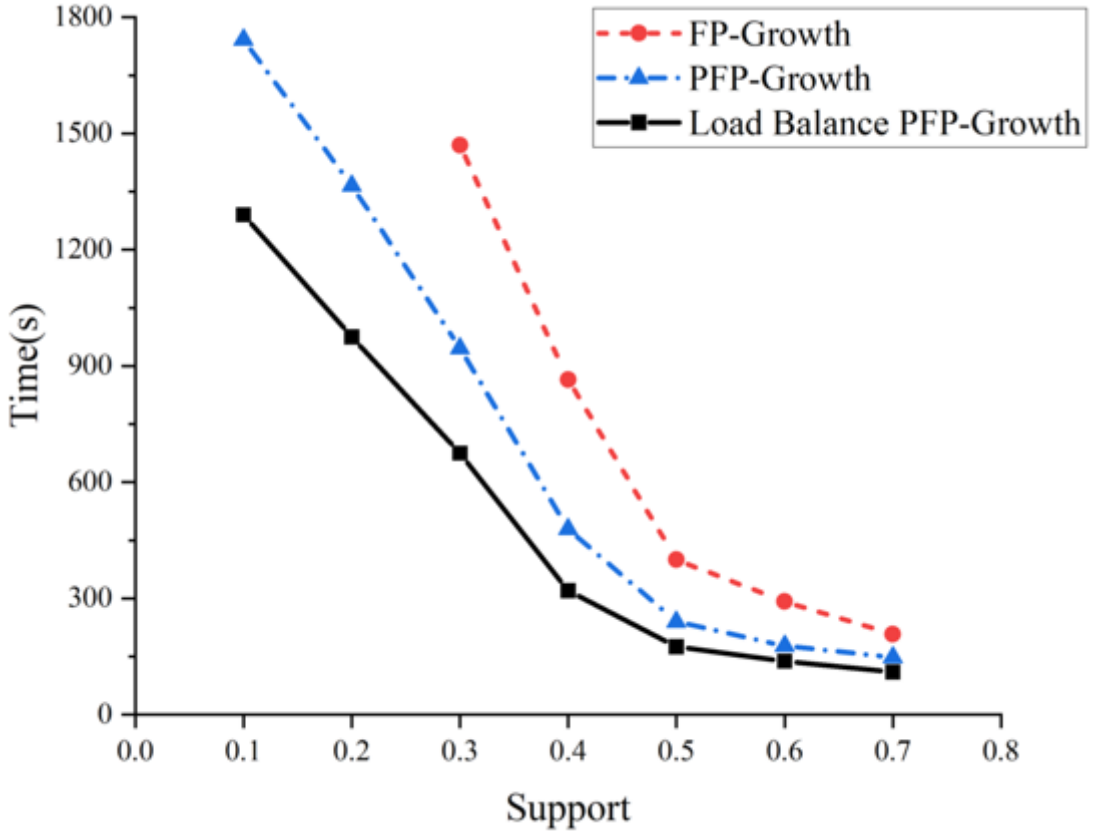

Figure 6

Comparison results of operation time with various minimum support thresholds