RESEARCH ON GIS MASSIVE TRAFFIC DATA ANALYSIS PLATFORM BASED ON HADOOP

Liu Jiayi[1],Yang Chengyong[1, *], Wang Peng[2], Ya Yunzhen[1]

1. Modern Educational Technology Center, Guilin University of Technology, No. 12 Jiangan Road, Guilin City, postcode: 541004, 2840313589@qq.com
2. Network Information Center, Guangxi Normal University, No. 15 Yucai Road, Guilin City, China. postcode:541004, 1172662867@qq.com

KEY WORDS: Traffic data, GIS, Hadoop, MapReduce

ABSTRACT:

In view of the limitations of storage and calculation of mass traffic data in traditional GIS platform, this paper uses efficient and scientific technical means to analyze the data, and proposes a Hadoop-based GIS mass traffic data analysis platform. The platform uses MapReduce as a distributed computing programming model to analyze massive data for urban traffic decision-making, and uses HDFS distributed file storage framework to store and manage massive traffic data at TB level or even PB level. Finally, the results are displayed by using geographic information system spatial visualization technology, and the impact of the data volume and the number of nodes in the cluster on the calculation time-consuming is analyzed and compared. The experimental results show that the use of distributed multi-node cluster can effectively improve the storage and computing efficiency of massive traffic data, and greatly accelerate the total task scheduling time.

* Corresponding author: Yang Chengyong；E-mail：ychy918@163.com;

Project fund: Promotion Project for Young and Middle-aged Teachers in Guangxi Universities（2018KY0252）

1. INTRODUCTION

With the rapid development of economy, people's quality of life has been improved, and urban car ownership has increased rapidly.The emergence of GIS, GPS and other spatial information technology can monitor the real-time operation process of the traffic system, and provide drivers with accurate road operation conditions and the best driving route. Traffic text, image, video and other data collected by various types of vehicle networks have an unprecedented explosive growth, and these data gradually show the characteristics of large amount of data, multiple data types, low value density, fast processing speed and increased complexity, namely "4V+1C" [1]. Due to the large amount of urban traffic data and high real-time performance, the current intelligent transportation system extracts traffic data in real-time through the card, GIS and other equipment, but the traditional data storage and processing technology no longer meets its needs [2]. Therefore, how to make the data structure and storage capacity flexibly expand, real-time accurate and efficient access to, upload, aggregate and store traffic data has become a major problem. Big data technology represented by Hadoop can provide more accurate data analysis results to realize real-time storage and calculation of massive traffic flow data.

At first, the most commonly used traffic flow theoretical models are car-following theory [3], vehicle kinematics model[4], cellular automata model[5], etc. Documents [6-8] elaborate traffic flow data mining algorithm more completely. Traffic flow data mining algorithm is mainly divided into frequent pattern mining algorithm, clustering analysis algorithm and classification analysis algorithm. Literature [9] proposes a method to integrate dynamic traffic conditions (DRC) such as traffic accidents into passenger sharing system to avoid unexpected delays caused by re-planning routes. Jiang Z et al.[10] proposed a Vehicle Cloud Computing (VCC) system. Because of the uncertainty of vehicle motion, task replication method was used to obtain the optimal strategy through value iteration in this system.

Because of Hadoop's powerful ability of storage and parallel computing, many scholars at home and abroad use Hadoop to realize the information mining and analysis of massive data. In order to deal with the imbalance of large data, López V et al. proposed a classification system algorithm based on fuzzy rules. The algorithm uses MapReduce framework to allocate the calculation operations of the fuzzy model. At the same time, cost-sensitive learning technology is added to the design to deal with the uncertainty introduced in the large amount of data. Ignore the learning of undervalued classes[11]. Sheng Zihao[12] aimed at the frequent and occasional traffic congestion problems, the cross-validation method was used to input information flow into vector classifier, and then the statistical method was used to distinguish frequent or occasional traffic congestion. Literature [13] proposes a large data mining method for gliding trajectory based on MapReduce. Combining MapReduce, a distributed computing framework based on Hadoop platform, with mining algorithm, the trajectory characteristics of taxis are extracted and analyzed. Chen Xiaobo et al.[14] proposed a Least Square Support Vector Regression (LSSVR) model based on sparse hybrid genetic algorithm optimization to predict short-term traffic flow.

Many traditional GIS traffic management and analysis systems at home and abroad are unable to store and calculate massive data in data centers effectively[15-17]. Therefore, this paper combines Hadoop's distributed computing and storage, geographic information system and database technology, proposes a Hadoop-based GIS massive traffic data analysis platform, which integrates collection, screening, storage, display and other functions, and will display the data on the browser side.

2. MAPREDUCE WORKFLOW IN HADOOP

MapReduce model based on distributed environment has considerable advantages in parallel processing. It not only considers how to schedule work mechanism to achieve load balancing, but also considers how to ensure smooth communication between data. Users do not need to consider how to implement MapReduce when coding. They only need

to understand Map mechanism which can divide data sets and Reduce mechanism which aggregates results. Therefore, the data file submitted by the user must be able to split into many blocks and complete its own computing tasks, as shown in Figure 1.

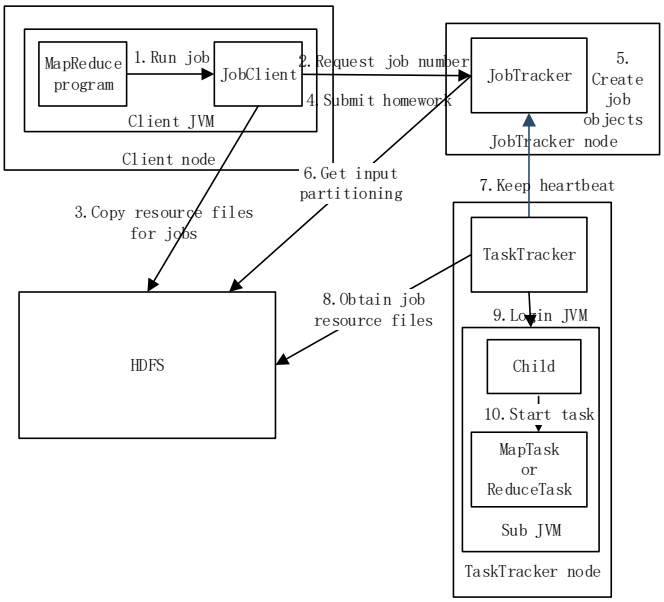

Figure 1. MapReduce workflow

Map-Reduce's job operation process steps:

(1) The client starts Map-Reduce and starts submitting jobs. The process is described as follows:

a. First, call the getNEWJobId( ) method in the JobTracker object to request a job ID number from JobTracker.

b. Check the output of the job. For example, if no specific output directory is specified or the output directory already exists, a related error message is returned and the job submission is stopped. Similarly, if the input directory does not exist or cannot be read, the corresponding error is returned and the job submission is stopped.

c. Segmenting the computing jobs and writing the segmented information into the job. split file.

d. Copy the required files such as JAR package files in MapReduce, configuration files and partitioning information obtained by client through calculation to HDFS.

(2) Job Scheduling: Operational migration without data movement

JobTracker receives messages, queues incoming jobs in a certain order, and schedules them dynamically in real time with different computational methods. When scheduling jobs, JobTracker creates tasks based on the hierarchy of input information, and their execution is sent to TaskTracker. For data localization only on the Map side, tasks are assigned to different processors, which are then copied to TaskTracker without data movement. Finally, Mapper and Reducer allocated to different clusters will also effectively address their diversity.

(3)TaskTracker sends a signal to JobTracker at intervals

The signal sent by TaskTracker tells the Map in real time whether it has completed the calculation process or not. In order to facilitate user queries and real-time control of the running status of the program, JobTracker sends a "success" signal after the last task in the queue is completed.

3 DESIGN OF GIS MASSIVE TRAFFIC DATA ANALYSIS PLATFORM BASED ON HADOOP

3.1 Architecture Design of GIS Massive Traffic Data Analysis Platform Based on Hadoop

The system is based on Hadoop system, including physical layer, resource pool, data acquisition layer, cloud storage and distributed computing layer and application interface layer. The whole platform structure is shown in Figure 2.

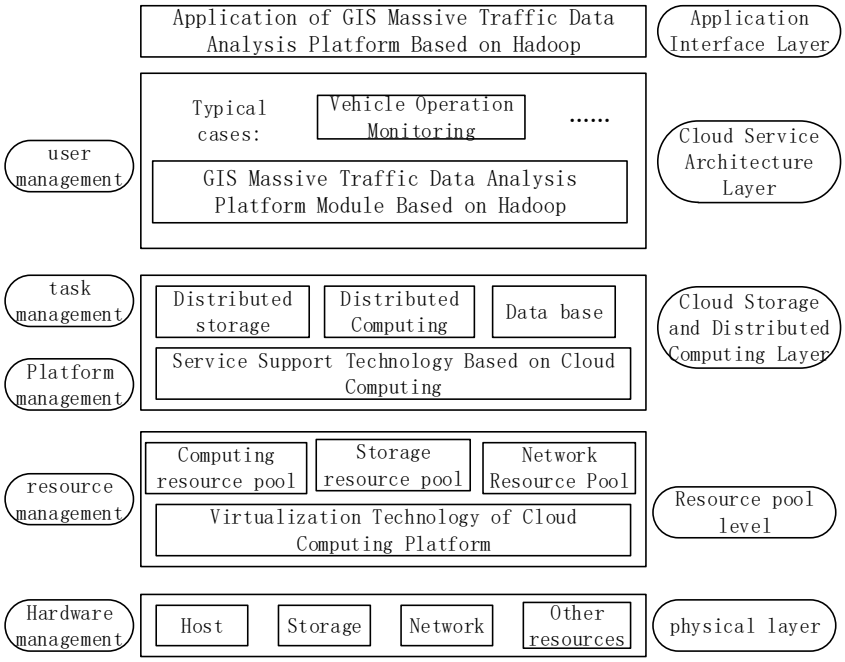

Figure 2. Hadoop-based GIS Massive Traffic Data Analysis Platform Architecture

Physical Layer: It is generally the underlying hardware device of cloud computing cluster, which is the IaaS layer composed of the same service nodes or simple physical devices. It is the basis of this platform, mainly including computing and execution, physical storage, network communication and other infrastructure.

Resource Pool Layer: Virtual all resources in a large container pool. When a request arises, the large pool will allocate a target resource and do a "busy" sign that the task is not allocated. It is transparent to the deployment and organization of the resources of the upper architecture.

Data acquisition layer: Web Driver and Pentom Js technology[18] are used to parse the content of the web pages and obtain the meteorological and geographic information needed. Because the information obtained is very messy, we use regular expressions to filter and discriminate these attribute information, and after format unification, we save it in Json text format.

Cloud Storage and Distributed Computing Layer: After formatting and storing massive traffic data from network crawling or real-time traffic data collected by the Internet of Things in the distributed file system HDFS, the MapReduce framework is parallelized and the final results are obtained by fast distributed computing.

Application Interface Layer: It mainly provides users with modules that can submit tasks and display data results. The front-end of the platform designed in this paper uses WebStorm development tools to visualize all pages. During the development process, all the data interaction of display pages is carried out with middleware Ajax. The interactive data format between pages and servers adopts JSON data format to ensure the timeliness and rendering effect of front-end pages.

3.2 MVC Technical Framework Design of GIS Massive Traffic Data Analysis Platform Based on Hadoop

In this paper, the overall design is mainly based on MVC[19] (Model-View-Controller) architecture. The Model (M) layer mainly manages the behavior and data of the application domain, and responds to the commands received from the control layer in a timely and effective manner. The data provided is converted into the required data format according to the set rules and provided to the view. (V) Layer display.The view layer mainly displays the data processed in the background in the Web interface. The data that the view layer needs to display comes from the model layer, and the M layer can map with multiple view layers at the same time. The control layer mainly handles various business requests submitted by users

through the set business logic rules. The control layer (C) mainly processes the data provided and notifies the view model to update its status, so that the model layer and the view layer can cooperate with each other. The goal of MVC pattern design is to reduce code reuse rate and achieve decoupling, so as to improve code readability and flexible application. MVC mode is mainly used to separate the interface display layer and the control layer perfectly, so that the design of the view layer is more concise and the operation efficiency is more efficient. The overall technical architecture of the platform is shown in Figure 3.

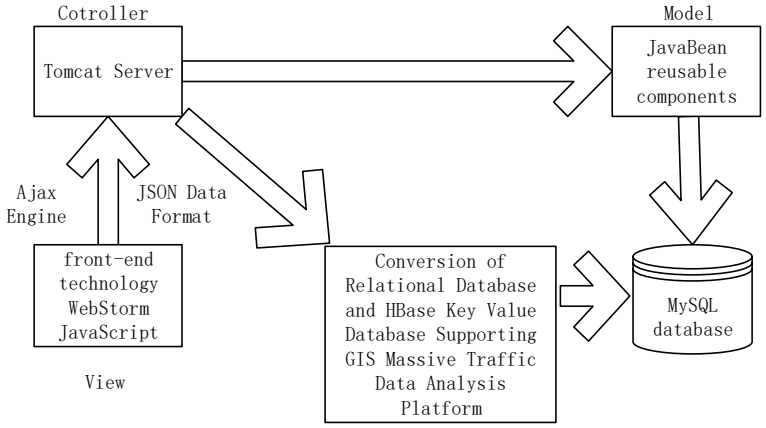

Figure 3. Technical architecture of GIS mass traffic data analysis platform based on Hadoop

3.3 Design of Travel Hotspot Module of GIS Massive Traffic Data Analysis Platform Based on Hadoop

In the system, Hadoop handles vehicle behavior mainly by calling related classes in Map-Reduce distributed framework and rewriting and compiling Mapper segmentation processing interface and Reducer merging grouping interface. The main implementation process is as follows:

(1) The corresponding format of data information to be processed is converted to the data processing format that Hadoop platform can support, and then the file reading class is called to read the data in batches.

(2)The corresponding rewriting of Mapper's segmentation processing interface includes the input and output types of vehicle's driving direction, passing time and license plate number, and the file types of intermediate data conversion.

(3) Rewrite the Reducer merged grouping interface accordingly.

(4)In the Job calling method dealing with the main function model, the corresponding parameters are set, including reading the passing record, outputting the license plate number and so on.Part of the code implementation is shown in Figures 4 and 5.

```
class Mapper extends TableMapper<Text,IntWritable> {
   private Text y = new Text();
   private IntWritable one = new IntWritable(1);
   private LinkedList<Car> cars = new LinkedList<Car>();
   publie void map(ImmutableBytesWritable rowKey, Result value,Context context) throws IOException,InterruptedException{
       String direction = Bytes.toString(value.getValue(Bytes.toBytes("lz1"), Bytes.toBytes("direction")));//方向编号
       String time = Bytes.toString(value.getValue(Bytes.toBytes("lz1"), Bytes.toBytes("time")));//过车时间
       String license = Bytes.toString(value.getValue(Bytes.toBytes("lz1"), Bytes.toBytes("license")));//车牌号码
       String key ="";
       if(cars.size()==0|| compareData(cars.get(0).getTIME(),time)) {//添加到cars中
          cars.add(new Car(direction,time,license));
       }else{
          Car car = cars.get(0);// 返回cars中的第一个元素
          cars.remove(0);//去除cars中第一个元素
          for(Car currCar : cars)
              if(car.getDIRECTION().equals(currCar.getDIRECTION())&&compareData(car.getTIME(), currCar.getTIME())) {//是否满足定义
                   if (car.getLICENSE().compareTo(currCar.getLICENSE())<0)
                       key = key + car.getLICENSE() +" " + currCar.getLICENSE();
                   else if (car.getLICENSE().compareTo(currCar.getLICENSE()) > 0)
                       key = key + currCar.getLICENSE()+" " + car.getLICENSE();
              }
          cars.add(new Car(direction,time,license));// 添加当前过车记录到cars
       }
   y.set(key);
   context.write(y,one);// Map的输出
   };
}
```

Figure 4. Mapper interface implementation code

```
class Reducer extends TableReducer<Text,IntWritable,Text,IntWritable>{
    private IntWritable result = new IntWritable();
    public void reduce(Text key,Iterable<IntWritable> values,Context context)throws IOException,Interruptedxception{
       int sum =0;
       for(IntWritable v :values)
           sum = sum + v.get();
       if(sum >= Global.MINSUP){
           result.set(sum);
           context.write(key,result);
       }
    };
}
```

Figure 5. Reducer interface implementation code

4. DISCUSSION

4.1 System Development Environment and Configuration

This paper mainly compares the performance of storage and calculation of mass traffic data between traditional Web GIS server and Hadoop cluster environment. The experimental data were screened for traffic data sets from many data sets of Lawrence Livermore National Laboratory (LLNL) [20]. Install and configure VMWare virtual software platform on physical machine with 8GB memory and 500GB hard disk, and then build Hadoop distributed cluster on this platform (6 virtual nodes are built in this paper). Using NAT network mode to communicate between nodes, each virtual node is configured as 1G RAM, 20GB external memory, 2.5GHz computing processor. The IP address of NameNode node in the Hadoop cluster is 192.168.138.138, the remaining five are DataNode nodes. The corresponding IP is 192.168.138.141, 192.168.138.142, 192.168.138.143, 192.168.138.143, 192.168.138.144 and 192.168.138.145. They are mainly responsible for the storage and management of data.

4.2 Implementation of GIS Massive Traffic Data Analysis Platform Based on Hadoop

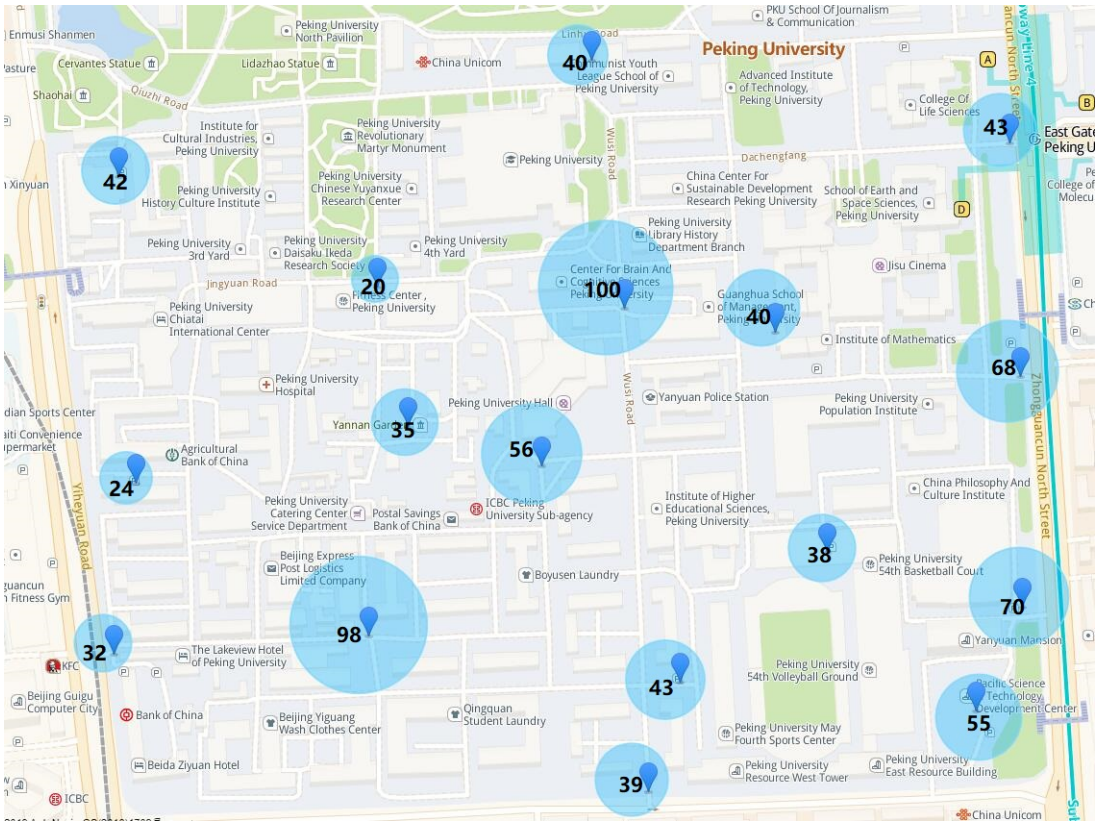

Figure 6. Analysis of Traffic Travel Hotspots

4.3 Computing and Storage Performance Testing of GIS Massive Traffic Data Analysis Platform Based on Hadoop

With the number of nodes changing, the calculation time of maximum, minimum and average of vehicle travel in the data set is shown in Figure 7. It can be seen that the computing performance of cluster increases with the increase of the number of nodes, but the more nodes, the more time cost of data transmission and communication, so the computing performance decreases with the increase of the number of nodes.

As shown in Figure 8, the storage performance of cluster increases with the increase of nodes. The most important thing is that if the amount of data exceeds the capacity of single-node server hard disk, it can not be stored. Cluster multi-node architecture can solve this problem.

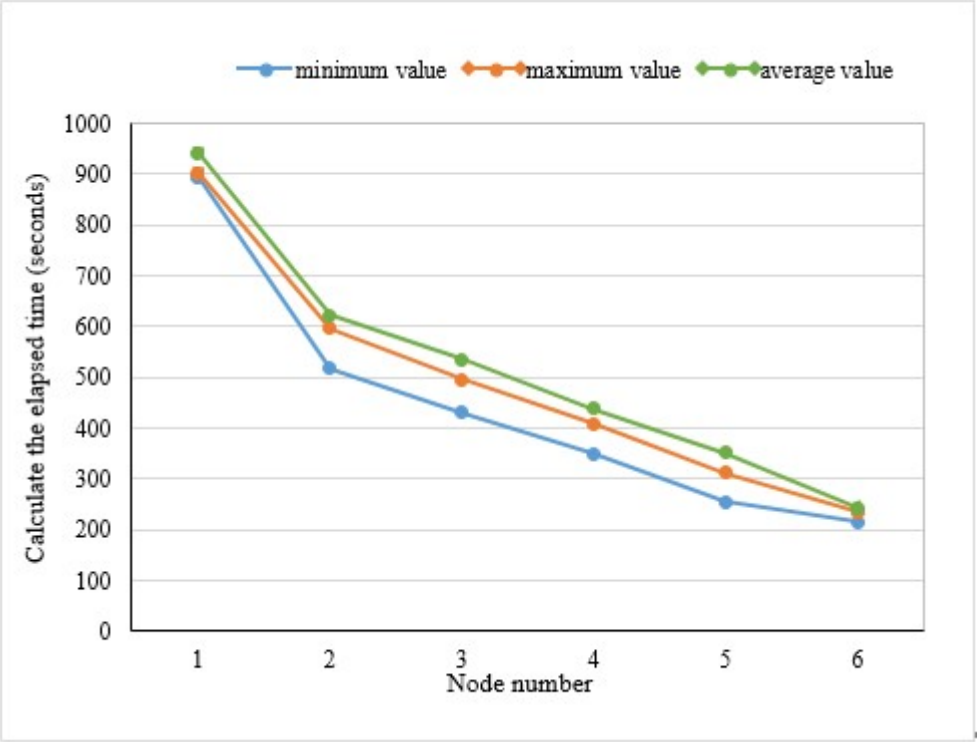

Figure7.Calculates the relationship between the elapsed time and the number of nodes

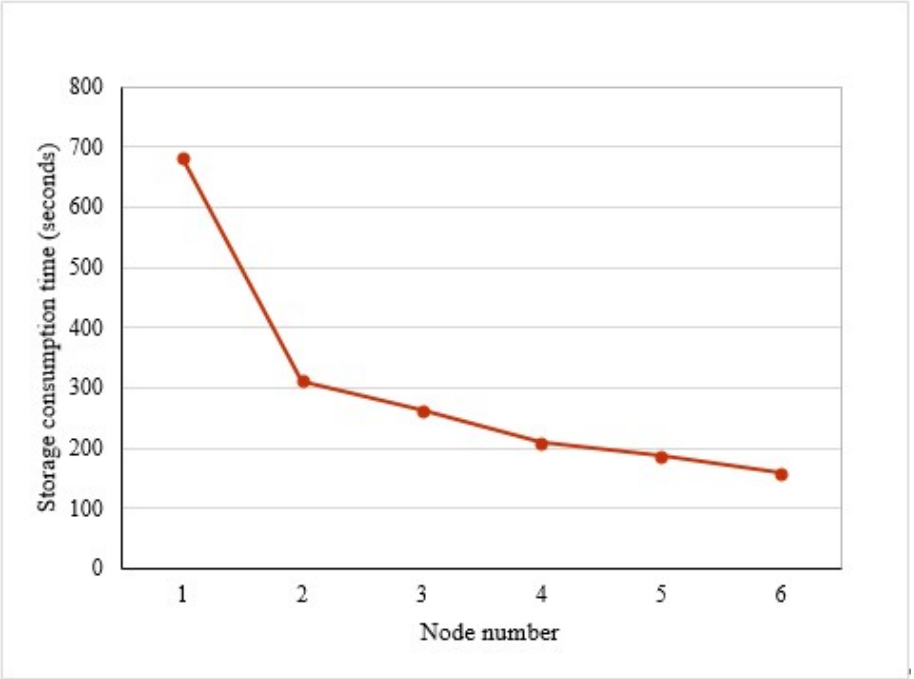

Figure 8 The relationship between storage consumption time and number of nodes

5. CONCLUSIONS

By analyzing the limitations of traditional GIS platform in massive data storage and calculation, this paper proposes a Hadoop-based GIS massive traffic data analysis platform. Cloud storage and cloud computing technology are used to solve the traditional single-node server constraints, and the impact of data volume and the number of nodes in the cluster on computing time-consuming is analyzed and compared. The experimental results show that the use of distributed multi-node cluster can effectively improve the storage and computing efficiency of massive traffic data, and greatly accelerate the total task scheduling time.

REFERENCES

[1] Ma J L, 2017.Problems and Solutions in the Application of Intelligent Traffic Data. *China Highway*,5, 102-103.

[2] Zhu L J.,2015.Distributed Storage and Analysis of Massive Urban Traffic Flow Data Based on Hadoop. *Yangzhou University*.

[3] Guo L W, Guo B J, Zheng H B, 2018.Review of Vehicle Following Behavior. *Shanxi Architecture*. 44 (21), 36-38.

[4] Xu J X, Xiong Z, Liu J Y, et al, 2017.Vehicle integrated navigation algorithm for smartphone platform based on vehicle motion model. *Chinese Journal of Inertial Technology*, 25 (02),203-208.

[5] Liu M , Shi J,2018.A cellular automata traffic flow model combined with a BP neural network based microscopic lane changing decision model. *Journal of Intelligent Transportation Systems*. 1-34.

[6] Han M, Ding J,2019.Overview of Frequent Pattern Mining in Data Stream. *Computer Applications*, 39 (03),103-111.

[7] Yang J R, 2017.A Survey of Cluster Analysis Algorithms for Data Mining. *Communication World*, 16, 291-291.

[8] Ramírez-Gallego, Sergio, Krawczyk B , García, Salvador, et al, 2017. A survey on data preprocessing for data stream mining: *Current status and future directions. Neurocomputing*, 239,39-57.

[9] Hargrave J , Yeung S , Madria S,2017.Integration of Dynamic Road Condition Updates for Real-Time Ridesharing Systems. IEEE International Conference on Mobile Ad Hoc & Sensor Systems. *IEEE Computer Society.*

[10] Jiang Z , Zhou S , Guo X , et al,2017.Task Replication for Deadline-Constrained Vehicular Cloud Computing: Optimal Policy, Performance Analysis and Implications on Road Traffic. *IEEE Internet of Things Journal.*

[11] López V, Río S D, Benítez J M, et al, 2015. Cost-sensitive linguistic fuzzy rule based classification systems under the MapReduce framework for imbalanced big data. *Fuzzy Sets & Systems*, 258(C),5-38.

[12] Sheng Z H.,2017. Research and application of traffic congestion identification and prediction algorithm based on data mining technology. *Qingdao University of Science and Technology*.

[13] Kong F , Lin X,2018.The method and application of big data mining for mobile trajectory of taxi based on MapReduce. *Cluster Computing,* 6,1-8.

[14] Chen X B, Liu X, Wei Z G, et al,2017. Research on short-term traffic flow prediction of road network based on GA-LSSVR model. *Transportation system engineering and information,* 17 (01),60-66+81.

[15] Duruz S, Flury C, Matasci G, et al., 2017. A WebGIS platform for the monitoring of Farm Animal Genetic Resources (GENMON). *PLoS ONE*, 12 (4),e0176362.

[16] Ahmadi M, Valinejadi A, Goodarzi A, et al,2017. Geographic Information System (GIS) capabilities in traffic accident information management: a qualitative approach. *Electronic Physician*, 9(6),4533-4540.

[17] Dong C S, Zheng W A, Ling X C, et al,2017. Study on Spatio-temporal Dynamic Segmentation Model in Transportation Geographic Information System. *Land and Resources of Shandong Province*, 5, 76-80.

[18] Zheng C L,2017. Research and Application of Internet Geographic Information Reptilian Technology. *Shandong Agricultural University*.

[19] Gao G, Wei H R, Li X D, et al, 2016. MVC Design Model. *Computer Knowledge and Technology*, 12 (1), 88-89.

[20] Xu M,2014. An Online Load Balancing Algorithm for Virtual Machine Allocation with Fixed Process Intervals. *Journal of Information & Computational Science*, 11(3),989-1001.