ATLANTIS PRESS

Advances in Engineering Research, volume 140

2nd International Conference on Electrical, Control and Automation Engineering (ECAE 2017)

A Novel Lightweight Middleware for Distributed Massive PMU Data Mining

Jianbo Yi*, Binbin Dong and Qi Huang

University of Electronic Science and Technology of China, Chengdu, Sichuan, China

*Corresponding author

***Abstract*—PMU data is facing with the problems of processing and application caused by the massive, distributed, multi-source, high redundancy existing in data. Therefore, this paper proposes a novel lightweight middleware to process and mine distributed massive PMU data. There employs light sampling parallel mining algorithm to deal with rules of data association, and proposes round robin algorithm to allocate computing tasks and data streams equally. Then the proposed middleware is applied in a multi-core computer. Test and comparison results from mining massive PMU data verify the feasibility and effectiveness of the proposed middleware technology in algorithm strategy and real-time requirement.**

Keywords-PMU data; light-eight middlerware; sampling parallel mining; round robin allocation algorithm

I. INTRODUCTION

Nowadays, the scale of modern power grid is expanding, and the structure and operation mode of power grid become more complex, so the difficulty of its measurement, analysis and control is increasing exponentially. The Wide Area Measurement System (WAMS) based on Phasor Measurement Unit (PMU) has been widely used in power system during the past decade [1, 2]. The PMU data contains great valuable state information of power grid, but there is difficult to achieve effective use with traditional means owing to the massive, distributed, multi-source, high redundancy of PMU data. At present, many researches like [3, 4] try to find new method or technology mining valuable information for power grid analysis and control from the massive PMU data, thereby how to avoid the phenomenon of data explosion but lack of knowledge is the urgent problem to be solved in wide area power grid monitoring and analysis.

Data mining, also known as knowledge discover in data, who aims at finding valuable hidden data in a large number of data sets, as an efficient technical method provides a feasible way for PMU data extraction. The tasks of data mining common include association rules mining, classification, clustering, and outlier item detection. Association rules mining is one of the most important tasks in data mining, whose purpose is to discover the implicit association between data items. The process of association rules mining focuses on how to find frequent itemsets in all data sets, which is also called frequent itemsets mining. Traditional association rules mining algorithms use iteration and counts to generate candidate frequent itemsets by statistics, like typical algorithm Apriori [5] and its improved algorithm in [6]. The classical Apriori algorithm needs to scan all the data sets many times, the candidate itemsets are too large, and it takes a lot of time to test the candidate itemsets. Aiming at the shortage of Apriori mining algorithm, research [7] proposes the algorithm based on the mining method of frequent pattern tree to extract frequent itemsets with memory decomposition. The data mining efficiency compared with the traditional algorithm has hundreds of times increasing, but the shortcoming of the FP-Growth algorithm is that it can only deal with small data sets and is powerless in the face of massive PMU data.

With the rapid development of information technology, the amount of data to be stored and analyzed has shown explosive growth, so the traditional association rules mining algorithm has been unable to meet the requirements of large data mining. The main difficulties are: a single computer cannot store all the excavated data and intermediate results during the mining process, the memory needed for the mining process is much more than that of a single machine, the computation time is too long to meet the time requirement. So many researchers begin to consider the solution of these problems through a distributed parallel computing environment. MapReduce is proposed by Google in [8], which is an easy to use and powerful parallel programming model. Its open-source middleware Hadoop has been widely used in many field of big data analysis to achieve a lot of traditional association rules mining algorithms migrating to the MapReduce model. The main idea of these algorithms is to use the distributed file system (HDFS) in Hadoop to solve the problem of massive data storage and use MapReduce to achieve parallel execution of mining algorithms [9].

This paper proposes a novel lightweight middleware to process and mine distributed massive PMU data. The efficient sampling mining algorithm first identifies the sample itemsets on the small amount of data to obtain the rules of data association of frequent itemsets. Then the round robin parallel algorithm allocates computing tasks and data streams equally with MapReduce design ideal. All the algorithms is integrated in the PMU data processing lightweight middleware, which has the characteristics of small memory resources and fast data mining speed.

II. MIDDLEWARE STRUCTURE DESIGN

According to the relevant standards of State Grid, the dynamic data upload rate of each PMU can be set to 25, 50, 100 times per second, while the PMU real-time communication rate should not be less than 19.2kbps and the bandwidth of upload channel not less than 2Mbps. The reality is that a large

number of PMU devices are distributed in wide-area power grid, so that the PMU uploaded data growth is very surprising. How to use the shortest possible time and the simplest equipment to deal with massive PMU data has become a major problem. If the strategy of PMU data mining employs cloud-computing platform, most of which are distributed processing framework, the requirements of hardware conditions are relatively high and communication time is much higher than data mining time. These defects are not conducive to improve efficiency and availability of the PMU data mining.

A novel structure of lightweight middleware to process and mine distributed massive PMU data is proposed in this section. The structure of middleware is divided by three parts shown in Figure I: input data interface, data mining algorithms and output data interface. The fact of input data interface is a raw data concentrator, which many distributed PMU devices send the measured data to sub-station, of cause sub-station data may come from a single PMU device or a plurality of PMU devices and other sub-stations.

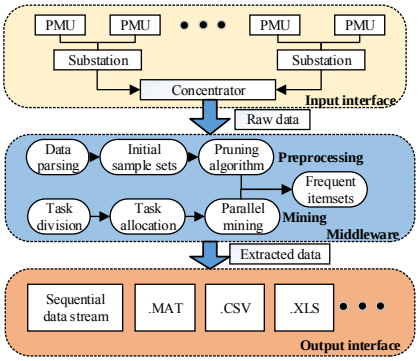

FIGURE I. THE STRUCTURE OF PMU DATA MINING MIDDLEWARE

Unlike the traditional distributed data processing system, the middleware utilizes single high performance computer to achieve parallel data mining technologies and all of these can be transplanted into the distributed system easily. Data mining algorithms in function include sampling parallel mining algorithm and round robin allocation algorithm. These data processing technologies can extract the valuable data out from massive PMU raw data. The third part is output data interface. Considering the problem of integrating other PMU data processing algorithm like time series analysis, the middleware need to provide conversion interface of extracted data to external application platform. While the data mining algorithm executes, the extracted data will be stored in different formats according to requirements of the follow-up analysis. The output formats supported by proposed lightweight middleware include sequential data stream, .MAT, .CSV, .XLS, ASC II text, etc.

III. PARALLEL DATA MINING ALGORITHMS

A. Sampling Parallel Mining Algorithm

The input PMU data is so massive and redundancy that extracting valuable information requires fast preprocessing of input data. The process called sampling mining algorithm mainly includes data frame parsing, frequent itemsets sampling and FP-Growth tree pruning algorithm.

The data frame of different PMU measurement devices is stored in corresponding position and set the device identification as column name. The device identifier includes the system area, the target equipment ID and the names of measured states, which represent such as positive sequence voltage amplitude, positive sequence voltage angle, A phase voltage amplitude, A phase voltage angle, B phase voltage amplitude, etc. So sampling mining algorithm needs to conduct the initial sampling method to achieve initial sample sets in real time database at first. Then use the pruning strategy based on FP-Growth tree [10] to cut out the unrelated data items to form smaller frequent itemsets.

When the PMU data set is loaded, the data set is divided into many blocks by single device ID and sequentially pressed into two-dimensional array. The index of a row in a two-dimensional array is set as the identification of the data layer shown in Figure II. If the user sets the data attributes that need to be extracted, the mining table is set by the first row of attributes relative to the first column.

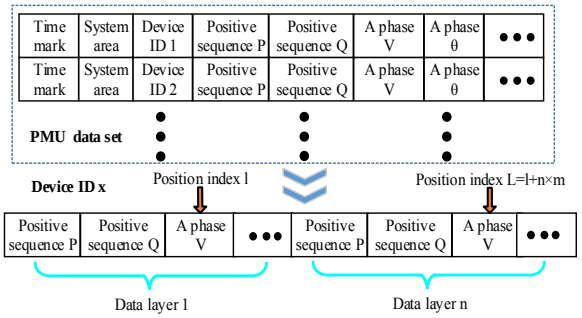

FIGURE II. PMU DATA SAMPLING MINING ARRAY

PMU data mining in this paper aims at extracting data with the same state name from different PMU devices. In the PMU data set, all identification information is stratified and identified by a single target device ID. Simple random sampling method will get several sets of continuous layer data, called group. The more the number of sampling groups and inclusion layers, the higher the accuracy of the mining table after the checkout.

This paper propose a pruning strategy filtering the irrelevant data sets from initial sample sets before analyzing to avoid this part of the data taking up processer time. The specific method of correcting the attributes in groups based on pruning method is to get the mining table and the number of groups as the following formula.

$$L=l+N\times m \tag{1}$$

Where L is the index of attributes set within the group, l is the index to set attributes in the mining table, N is the number of layers contained within the group, and m is the number of data in the layer.

The algorithm only validates whether the index of the first attribute or the last attribute in the group satisfies the above formula. If it satisfies, it means that the mining table is applicable to this group. If not, the group is divided into two small groups according to the level and it is verified again until

a certain level of mining strategy is not conformed to the mining strategy. After getting the applicable mining formula, the itemsets that does not need to be considered is cut off. The pruning method will destroy the upload PMU data set, only keep the extracted data table and free the memory.

B. Round Robin Allocation Algorithm

For the process of massive data mining, the most basic factor is the efficiency of mining algorithm and its time cost. This section studies task parallel allocation and load balancing parallel mining algorithm based on MapReduce technology. Shown in Figure III, the Map method of MapReduce model maps the key value pairs constructed from the original data to new key value pairs, which realizes data partition according to a certain attribute and makes it suitable for distributed processing. The concurrent Reduce method is used to ensure that the key value pairs of all the maps are merged together to output results [11].

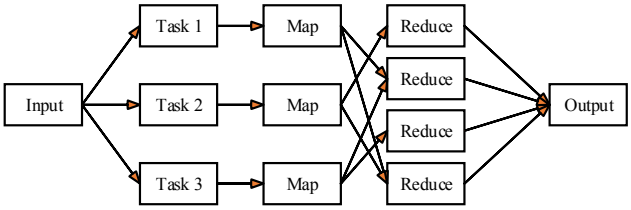

FIGURE III. MAPREDUCE TASK DIVISION

Based on the idea of MapReduce parallel processing, the effective partition of data is very important in order to ensure the load balancing of processing nodes. Shown in Figure IV, the algorithm sequentially receives multiple data frames, and stores them in a ring memory buffer with n as a group. The starting mark is set up at the first group of data that is not processed. When the buffer is about to overflow, the algorithm will create an overflow file in local file system to write the data at the beginning of the buffer. After n data frames merging, pruning mining algorithm continues to process several group data mining processes, then extracts results from the memory into the output interface and destroys the processed data and updates the starting mark.

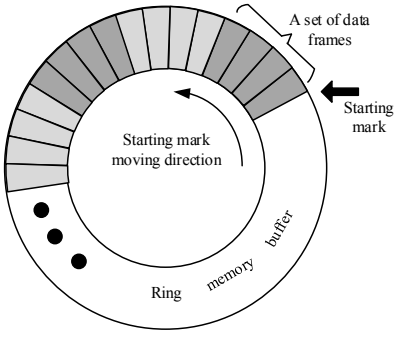

FIGURE IV. RING MEMORY BUFFER STRATEGY OF DATA

During the entire mining process, the buffer continuously receives the updated data, does not interrupt, and sets the "empty", "not full" and "full" mark information. This strategy can effectively reduce frequent access to I/O and using parallel mining algorithm can improve the mining speed very well.

Considering the parallel processing nodes should receive the tasks equivalent to their processing capabilities after data partitioning, the round robin allocation algorithm of static distribution is employed in Figure V. The algorithm puts the entire running job into a queue according to the FCFS strategy, then sets certain time slice and assigns the time slice to the team's first job each time. If the job is finished and time slice is not used yet, remove the job from the running queue and assign a new time slice to the next job. If time slice is used up but job is not finished, the job is inserted at the end of the ready queue waiting for scheduling.

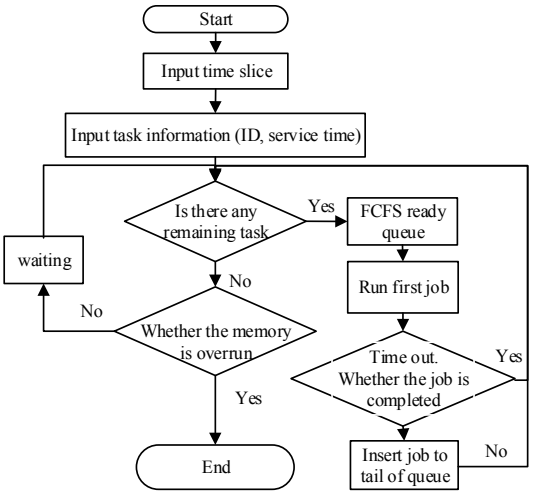

FIGURE V. ROUND ROBIN ALLOCATION ALGORITHM

According to the characteristics of PMU data, the algorithm sets time slice size and establishes operation information including input task ID, input service time and run time. The job is sorted by the time of arrival and saved as the ready queue. When a job is mapped to a processing unit, it will first confirm whether the buffer is "full", and if so, the overflow file will be processed first. When the buffer is changed to "not full", the pruning strategy is carried out by the group data at the starting mark in the ring memory buffer. Each process starts as an established task, since there is no overlap or default in the assignment of tasks, so there is no need for communication between the various processes. The extracted data after mining algorithm is stored in memory as form of an array. When memory is full, send interrupt task to the queue and arrange a process out of data mining to deal with data storage, so it can make full use of the process to perform data mining tasks.

IV. CASE STUDY

The parallel data mining and processing algorithms of lightweight middleware of massive PUM data are presented in the above description. In this section, the feasibility and effectiveness of the proposed algorithms are verified through actual massive PMU data processing test. The configuration of test computer is Intel CPU 3. 90GHz, RAM 8.0GB, (data mining available 7.0GB), 4-core processor and Windows OS.

Study case: data source is 700MB massive PMU data, including 1390 target devices in Sichuan, Tibet and cross regional power system, total about 28000 data sets, each set saved 3000 frame data. The data mining algorithm embedded in middleware is used to mine "positive sequence P", "positive sequence Q", "power factor" etc. 13 attributes data sets. The sampling mining algorithm set the mining table index as (1, 2, 3, 4, 5, 6, 9, 10, 11, 12, 17, 18, 19). There are two parallel data mining algorithms with Python compared in following data test, one is FP-Growth algorithm, and the other is FP-tree pruning algorithm proposed in this paper. The data mining rate comparison result of these two kinds of mining algorithms is shown in following Figure VI.

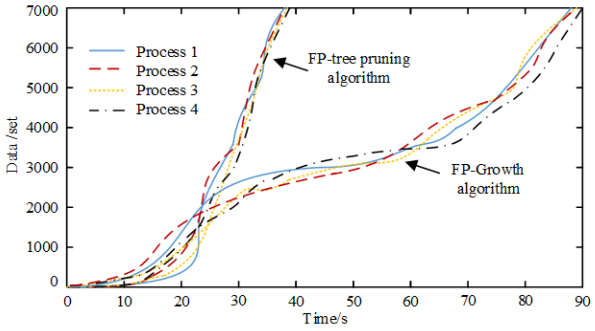

FIGURE VI. RATE COMPARISON OF TWO PARALLEL DATA MINING ALGORITHMS

Comparing two kinds of parallel data mining algorithms, there is little difference in the data load assigned to each process, which shows that the round robin allocation algorithm has good effect to achieve load balancing. When it comes to the problem of mining rate of two algorithms, the proposed FP-tree pruning algorithm is obviously superior to the traditional FP-Growth algorithm. However, at the beginning of the processing, the proposed algorithm does not make up the advantage. The reason is that the process of sampling mining needs more complicates data preprocessing, including data reconstitution like Figure II, setting mining table with formula (1) and pushing data frames to ring memory buffer as Figure IV.

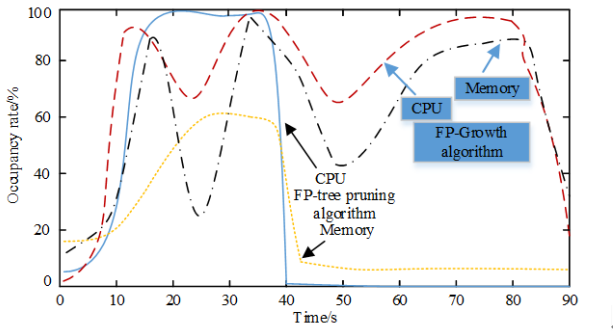

FIGURE VII. COMPARISON OF RESOURCE OCCUPANCY

Figure VII shows the resource usage of the two algorithms at run time. There are nearly 100% CPU occupation rate after two algorithms starting to mine, but the CPU occupation rate of traditional FP-Growth algorithm has several gaps at running time. This is the inherent defect owing to that the massive data can not be loaded into memory at one time, so the algorithm has to deal with I/O operations between RAM and external memory frequently. The proposed FP-tree pruning mining algorithm employs sampling table and ring memory method to ensure less I/O operations, and used data destroyed timely.

The above test and analysis show that the proposed lightweight middleware installed in ordinary PC is able to achieve massive PMU data mining. There are good performance and efficiency with parallel data mining algorithms embedded in middleware, which improve PMU data preprocessing speed up to 70MB/s and data mining speed up to 25MB/s. This middleware can meet the requirement of real-time measurement information processing of power system.

V. CONCLUSIONS

This paper proposes a novel lightweight middleware to process and mine distributed massive PMU data. The middleware propose using sampling parallel mining algorithm and round robin algorithm to design parallel data mining strategy. The results of test case verify that the middleware installed in multi-core computer has good feasibility and efficiency to mine distributed massive PMU data. These technologies are likely to be widely used in WAMS or other distributed information collection systems.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of the Technology Research and Development Program of Sichuan Province, China (2017GZ0054) and the Fundamental Research Funds for the Central Universities (ZYGX2016J142).

REFERENCES

[1] S. Kai, S. Likhate, V. Vittal, and V. S. Kolluri, "An online dynamic security assessment scheme using phasor measurements and decision trees," IEEETrans. PowerSyst. ,vol.22, pp.1935–1943, 2007.

[2] C. Liu, Z. Chen, C. L. Bak, and Z. Liu, "Adaptive voltage stability protection based on load identification using Phasor Measurement Units," inProc. 2011Int. Conf. Advanced Power System Automation and Protection (APAP), pp. 1246–1251, 2011.

[3] M. He, V. Vittal, and J. Zhang, "Online dynamic security assessment with missing PMU measurements: A data mining approach," IEEE Trans. Power Syst., vol.28, pp.1969–1977, 2013.

[4] T. Guo, V. Jovica, "Online Identification of Power System Dynamic Signature Using PMU Measurements and Data Mining," IEEE Trans. Power Syst., vol.31, pp. 1760 – 1768, 2016.

[5] R. Agrawal, R. Srikant. "Fast algorithms for mining association rules in large databases, " Proc of International Conference on Very Large Data Bases. Morgan Kaufmann Publishers Inc. pp.487-499. 1994.

[6] S. Brin, R. Motwani, D. J. Ullman, et al. "Dynamic itemset counting and implication rules for market basket data," ACM SIGMOD Record, vol.26 (2),pp. 255-264, 2001.

[7] R. Chang, R. Liu, "Apriori safety improvement electronics and Al gorithm," 2011 Conference on international optoelectronics. pp. 1476-1478, 2011.

[8] J. Dean, S. Ghemawat, "MapReduce: Simplified data processing on large clusters," Operating Systems Design Implementation, vol. 51(1), pp. 147-152, 2004.

[9] K. Shvachko, H. Kuang, S. Radia, et al. "The Hadoop distributed filesystem," Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST), pp. 1-10, 2010.

[10] Z. Kun, H. Rui, Z. Na, "Efficient frequent patterns mining algorithm based on MapReduce model," Journal of Computer Science (in Chinese), vol.44(07), pp.41-37, 2017.

[11] Z. Xuejun, Z. Hao, Y. Jingtong, "Research on RM scheduling algorithm based on coprocessor and dynamic time slice," Computer Technology and Development (in Chinese), vol. 25(03), pp. 188-192, 2015.