Review Article

Advances in Machine Learning & Artificial Intelligence

Blockchain Towards Prioritization-Based Distributed Storage of Big Data for Internet of Vehicles

Dan Ye*

Department of Computer Science & Information Engineering, National Taiwan University, Taipei 106, Taiwan

***Corresponding author**
Dan Ye, Department of Computer Science & Information Engineering, National Taiwan University, Taipei 106, Taiwan

Submitted: 26 Jan 2022; **Accepted**: 03 Feb 2022; **Published**: 10 Feb 2022

Citation: *Dan Ye. (2022). Blockchain Towards Prioritization-Based Distributed Storage of Big Data for Internet of Vehicles. Adv Mach Lear Art Inte, 3(1), 23-26.*

Abstract

This paper proposes a prioritization-based distributed storage of big data processing application in Internet of Vehicle (IoV) system. Designing a scalable, high-performance big data distributed storage system for IoV, an advanced data-processing system for car services. The novel contribution focused on developing vehicular multi-channel control protocol that control the prioritization of services, according to bit rate, transmit power, speed, inter-vehicle distance. The proposed scheme can achieve higher performance in IoV storage system.

Index Terms- *IoV, sensor fusion, distributed storage, prioritization, edge computing, cloud computing.*

Introduction

Fog computing fuses cloud, network, and fog. Automatic sensor fusion, sensor data from vehicle networks store above the cloud using distributed deep neural networks (DDNNs) to improve object recognition accuracy.

Designing large-scale real-time Internet-of-Vehicles (IoV) systems to process distributed storage of big data and provide data-driven services. The IoV data includes high-frequency vehicle status, location information, driver activity (bit rate, transmit power, speed), inter-vehicle distance, service priority and passenger-trip information. This paper develops a scalable IoV system for processing big data in car services. Big data technologies process large-scale driving-data streams. The vehicles are connected to the cloud data center and upload vehicle statues to center.

The cloud collects and analyzes data, sends the value-added information back to vehicles. Cloud-based IoV or fog-based IoV system in vehicle networking process a small stream of data or a stream of big vehicle data, supporting various data-centric services. The first design feature of distributed system is high scalability, which can deal with big scale of data. Real-time and accuracy is second requirement in processing large-scale data. Highly reliable IoV system supports safety-critical services, is third characteristic.

The remainder of this paper is structured as follows: Section 2 formulate new strategy on distributed storage of big data for IoV. Section 3 elaborates prioritization-based multi-channel control scheme. A summary concludes the paper in Section 4.

New Storage Strategy

In this paper, designing a scalable, high-performance big data distributed storage system for IoV, an advanced data-processing system for car services. It has multiple data sources including vehicle status data (such as speed, transmit power, bit rate, inter-vehicle distance), driver activities (such as starting a service, service priority, picking up a passenger). New IoV big data system has the ability of distributed processing and data storage.

The backend servers in cloud-centric vehicular network collects, analyzes, stores the data uploaded by vehicles. A high-performance big data cloud-centric IoV storage system requires solving two aspects of problem. The first aspect is data separation. Identify cloud data and fog data in real-time environment, then separate the cloud data with fog data and out them into different storage platforms with different performance. The second aspect is preprocessing. Big dataset scan for database is painful, but data appending can be fast. Extract a small dataset with a higher density of value by using stream processing. Separate this part of data with the raw data. Storage subsystem can provide high throughput for data write, and high performance for the queries that are based on the small processed data. Furthermore, put some cloud data in in-memory cache to achieve a real-time data access [1]. To design a data management subsystem, three different storage platforms are adopted in IoV system, including in-memory caching, relational database, big data storage. There are multiple selections for each kind of storage platform. Car Query API is an easy-to-use JSON

based API for retrieving detailed car and truck information, including year, model, trim, and specifications. Write own JavaScript or server-side API interactions to use the available data. Car Query source vehicle database contains the exact vehicle data that is used to power the Car Query API. CSV & MySQL file containing extensive data on all available models. Detailed specification includes engine location, engine type, engine cylinders, engine displacement (cc), engine displacement (l), engine displacement (cubic inches), Engine Bore (mm), Engine Bore (in), Engine Stroke (mm), Engine Stroke (in), Engine Valves Per Cylinder, Engine Valves, Engine Max Power (HP), Engine Max Power (PS), Engine Max Power (kW), Engine Max Power RPM, Engine Max Torque(Nm), Engine Max Torque (Lb-Ft), Engine Max Torque (kgf-m), Engine Max Torque RPM, Engine Compression Ratio, Engine Fuel Type, Drive, Transmission Type, Top Speed (KPH), Top Speed (MPH), 0-100 kph (0-62mph), Doors, Seats, Weight (kg), Weight (lbs), Length (mm), Length (in), Width (mm), Width (in), Height (mm), Height (in), Wheelbase (mm), Wheelbase (in), Fuel Economy City(l/100km), Fuel Economy City(mpg), Fuel Economy HWY(l/100km), Fuel Economy HWY(mpg), Fuel Economy Mixed(l/100km), Fuel Economy Mixed(mpg), Fuel Capacity(l), Fuel Capacity(g).

Many NOSQL database, such as Cassandra, HBase, and MongoDB, can be adopted. HBase is a good candidate as archive storage platform for raw data. HBase can easily integrate with Hadoop to achieve big data batch processing ability. HBase provides a high-performance key-value storage, which perfectly fits the requirements of storing the raw vehicle data. The raw vehicle data includes vehicle status, vehicle trajectories, driver activities related data, and user-order related data. Those data use ID (vehicle ID, user ID, driver ID) and timestamp as the index key. Raw driving data is managed with RDBMS such as Mysql and Postgresql. Generally, <ID, timestamp> is labled as index. Relational databases must maintain huge index files to index the big dataset. As a distributed key-value storage, HBase has the scalability with data volume. Memcached and Redis are in-memory caching platforms. They provide high-performance key-value storage. Memcached provides high-performance on key-value based caching. IoV storage system adopts Memcached to be data buffering and high-performance data storage media. IoV system database has more data types, data accessing functionalities.

Window is querying with Alibaba cloud RDS for PostgreSQL. One of the biggest issues of IoV is collecting vehicle's travel tracks in real time. Several track records may be accumulated or reported at intervals. There are many platforms that can be used as data bus, such as MetaQ, RocketMQ, ActiveMQ, and RabbitMQ.

Provide highly dependable services, IoV storage system can be separated into three layers, the infrastructure layer, which includes the cluster and operating system; the computing platform layer, which includes the processing platforms such as Hadoop, Storm; and the application layer. The infrastructure layer and computing platform layer provide useful information for assuring service dependability. The applications in IoV system run permanently with the streaming data continuously comes.

Prioritization-based multi-channel control protocol

This paper develops vehicular multi-channel control protocol that control the prioritization of services, according to customer preferences to choose a particular service [1]. The concept of high-priority services, low-priority services is fused into channel allocation. Service prioritization is first applied into vehicular multi-channel concurrent transmission. To efficiently utilize the spectrum, vehicles operate on multiple channels simultaneously. Service prioritation is used to alternate between a control channel (CCH) and service channels (SCHs) for single-radio transceivers, and between SCH for dual-radio transceivers [2]. Multi-channel operation can improve the spectral efficiency. Platoon has longitudinal control which can adjust speed and inter-vehicle distance. When vehicular multi-channel concurrent transmission, the priority of vehicular channel access can be designed in details. The core thought is critical awareness priority combined with vehicular channel service control. Idle or busy channel access requires high priority or low prioirity which provides different access control services [3].

This paper proposed four schemes for adjusting adaptively the priority of service channel access.

1. Bit rate-based prioritization of service
2. TX power-based prioritization of service
3. Speed-based prioritization of service
4. Inter-vehicle distance-based prioritization of service

Different channels have concurrent transmission to improve channel efficiency. Shared channel can be designed to supply service. Concurrent services exist in shared channel. Different channels implement different transmission service. According to different vehicle groups, the priority of channel service can be configured.

A shared in-memory database has high performance, end-to-end performance evaluation of entire stream-processing procedure from data receiving to data query. The processing subsystem is deployed on a virtualized cluster. A dedicated server, HBase and HDFS are deployed in a non-virtualized environment with three 12-node clusters. The total capacity of HDFS is above 600TB.
IoV system can easily handle the data stream uploaded by 30,000 vehicles. The total processing throughput of IoV system has reached 240,000 data instances per second (each data instance is around 100KB).

Use real-time vehicle tracking as test data, data are uploaded from vehicles and go through a series of platforms including the buffering platform, stream processing platform, in-memory caching platform, and being queried by the web server.

Compare the timestamps between the data being generated and data being processed, record the length of the queue of computing tasks. Queue length reflects the processing delay.

According to scalability, low latency, high throughput, high reliability, and high availability, data-storage subsystem needs different priority services to improve storage performance. Core data storage system maintains real-time large-scale batch processing, the combination of online with offline processing, achieving real-time performance. Data separation and preprocessing are required to improve throughput for IoV data storage. HBase is typical NOSQL database, key-value storage integrated with Hadoop to achieve big data batch processing performance. It can store raw vehicle data including vehicle status, vehicle trajectories, driver activities, user-order related data. Those data use ID (vehicle ID, user ID, driver ID, service ID, priority ID) and timestamp as the index key.

VMware Virtual SAN provides high performance, enterprise-class shared storage [4]. The performance benefits of high throughput and low latency, and linear scalability.

Building Intel architecture-based platform increases storage capacity. Collect, process, store and analyze vehicle terminal data in real-time, and support real-time access from a large number of end points. Enable a safe, stable, and complete information service solution for the Internet of Vehicles [5]. Deploy servers based on Intel Xeon processor E5-2600 v2 and E7-4800 v2 product families running on VMware vCenter as the cloud computing architecture. Servers based on Intel Xeon processor E7-4800 v2 product family harness real-time data processing. Servers based on Intel Xeon E5-2600 v2 product family running an Apache Hadoop/Hbase big data platform utilize offline data analysis. Deploy a software-defined and distributed storage architecture using Intel Solid-State Drive (Intel SSD) DC S3700 Series and Intel Ethernet Converged Network Adapter X520 to provide highly reliable data storage and flexible expansion on a 10GbE network to improve data throughput and network performance.

Through cloud computing and big data technology, data storage and analysis can be improved. Seen from Figure 1, big data in sensors with higher priority has been stored distributed firstly. With service ID, data in IoV has been divided into four categories. Service ID 1-7 defines as priority 1, service ID 8-14 defines as priority 2, service ID 15-21 defines as priority 3, service ID 22-28 defines as priority 4. Priority 1 has highest priority. Lower-priority services employ single channel operation while higher-priority services employ multi-channel operation. According to these sensors application priories, distributed storage system makes use of such application identifiers.

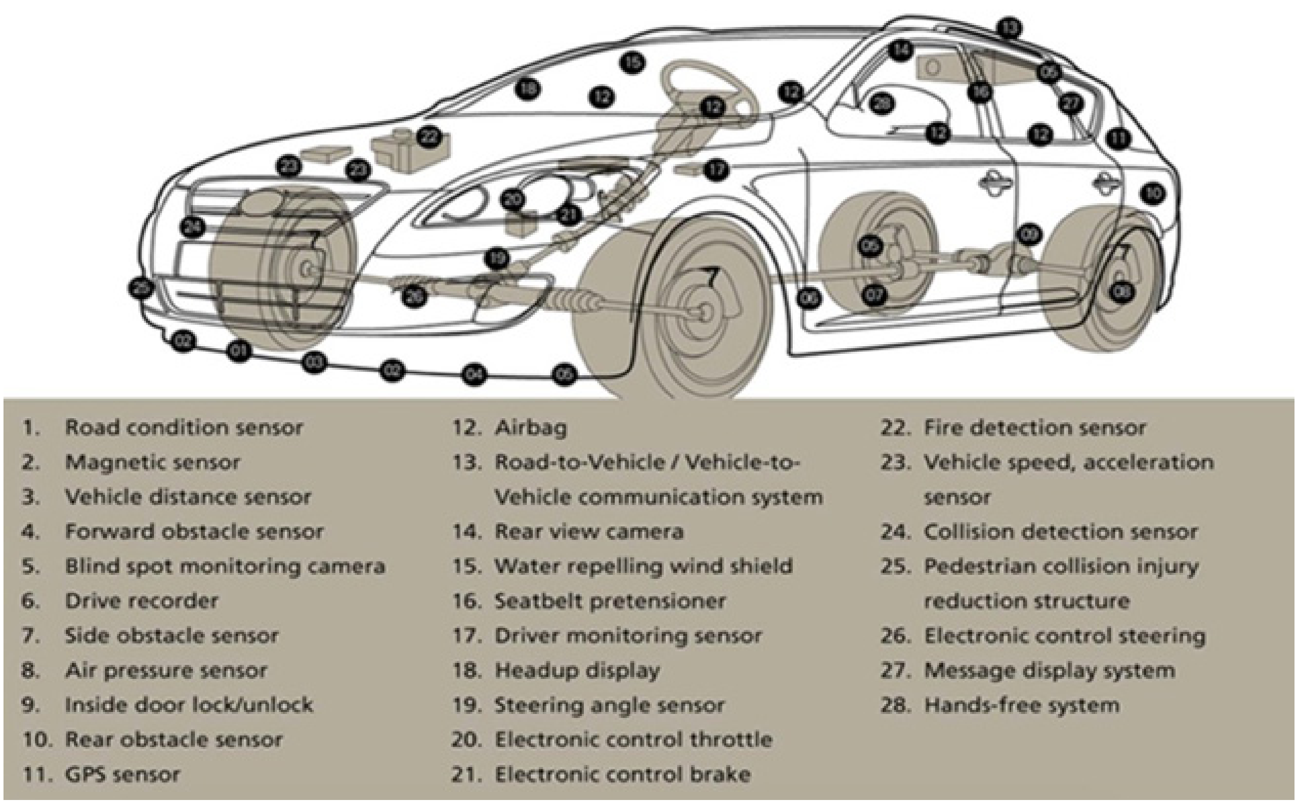

Figure 1: The sensor deployment of IoV system

According to vehicle ID in database optimization for IoV, data gathered for different vehicles is usually stored into a database. If no optimization is made, after entering the database, the data of different vehicles may be staggered. That is, the data of different vehicles may be stored in one data block. A lot of data blocks will be scanned (scanning IO amplification) when the track of a single vehicle is queried. There are two optimization methods to speed up the querying process. The main purpose of two methods is to reorganize the data based on query requirements to decrease scanning IO.

1. Store into the database after the terminal gathers grouping and sorting. For example, after receiving the data submitted by the vehicle, the program groups vehicle IDs, divides service priorities, sorts them by time, and stores them into the database. In this way, the data of the same vehicle in same priority will fall into the same data block.
2. Use a partition to reorganize the data in the database. For example, data can be stored based on the vehicle ID, or vehicle HASH partition.

Conclusion

This paper presented a new storage scheme for multi-channel control protocol in IoV system. For efficient vehicle communication in cloud platform, priorization-based service channel access scheme is proposed. High throughput, low latency, high reliability, enhanced storage performance benefits from high-priority services. Data separation and preprocessing, reorganizing accerlerate database optimization for IoV.

Reference

1. Zhang, M., Wo, T., Xie, T., Lin, X., & Liu, Y. (2017). Carstream: an industrial system of big data processing for internet-of-vehicles. Proceedings of the VLDB Endowment, 10(12), 1766-1777.
2. Boban, M., & Festag, A. (2016). Service-actuated multi-channel operation for vehicular communications. computer communications, 93, 17-26.
3. Ouaddah, A., Abou Elkalam, A., & Ait Ouahman, A. FairAcces: a new blockchain-based access control framework for the Internet of Things. Secur. Commun. Netw. 9 (18), 5943–5964 (2016).
4. Intel. Improving storage agility with VMware virtual SAN and Intel Architecture, Solution Brief, 2016.
5. Intel. Data center architecture for Internet of Vehicles, 2015.