A Real Time Data Association Prototype System for Multi-Tenants in Big Data

Ge Fu, Yu Ding, Siyu Jia

National Computer Network Emergency Response Technical Team/Coordination Center of China, Beijing, China
Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
University of Chinese Academy of Sciences, Beijing, China

Abstract. With the development of multi-source heterogeneous data production and multi-application multi-tenant data consumption, the requirement of data association analysis is growing. Current data association analysis technology cannot satisfy the requirements of streaming data analysis and processing in a real scenario. In this paper, in order to achieve the real-time data association analysis, firstly, we solve the key problems of streaming data association; secondly, we construct a data real-time association computing system for multi-tenants for a Kafka-based big data analysis platform. The system has characteristics of high availability and high efficiency.

1. Introduction

With the development of multi-source heterogeneous data production and multi-application multi-tenant data consumption, more and more data association analysis is needed. Technologies of data association analysis can be roughly divided into three categories. Firstly, the association between offline data, i.e. offline data association for data which is stored into “Hive”[1]. Secondly, association between offline data and real-time data: In common sense, it can be used for real-time streaming data LEFT JOIN an offline table. Thirdly, the association between real-time data: i.e. two different streaming data do an operation of “JOIN” to produce a new streaming data. Current data association analysis technology can only support the first two types of data association requirements, current data association analysis technology cannot support real-time association analysis between streaming data. For the second type of requirement, the general approach is to use “Spark Streaming” [2] or “Storm”[3] components to process streaming data and offline data association analysis after the data being subscribed from “Kafka”. The problem with this architecture is that multi-tenants require to subscribe to large amounts of data in order to solve the problems of data association. As shown in the left half of Figure 1, all applications need to subscribe to each Topic that needs to be associated, i.e. one data is subscribed n times by different consumers, which wastes a large amount of network bandwidth and data processing resources (memory of components such as Spark Streaming, Hive, I/O, etc.). Therefore, in order to reduce costs of data association analysis in the context of multi-application and multi-tenant, a better solution is to “move forward” the operation of association calculation and analysis, which means doing operations of the association calculation and analysis in the layer of Kafka (as shown in right half of Figure 1).
In this paper, in order to achieve an effective real-time data association analysis, firstly, we solve the problem of association streaming data; secondly, we construct a data real-time association computing system for multi-tenant. The system has a character of high availability and high efficiency.

Figure 1. “Moving forward” the operation of association calculation and analysis

2. Related Work

In 2017, the Apache Foundation released a key project “KSQL” [4], which is a Kafka based streaming data manipulation engine. It provides an easy-to-use, yet powerful interactive SQL interface for stream processing on Kafka, without the need to write code in a programming language such as Java or Python. KSQL is scalable, elastic, fault-tolerant, and it supports a wide range of streaming operations, including data filtering, transformations, aggregations, joins, windowing, and sessionization. On February 28, 2018, Spark 2.3.0 was released, with significant enhancements to streaming data operation, which supports “Streaming Join” [5]. Flink1.5.0 was released on May 28, 2018. For this version, streaming data operations was enhanced and streaming complex operations was added [6]. In 2017, the group of Ge Fu released a prototype system for real-time data operation system. It can do the operation such as filtering and injecting for a single streaming data [7]. These systems have a similar function for streaming data operation compared to our system.

3. Key Technologies of Streaming Data Association

For real-time arrival of data streams, the streaming data processing system must quickly respond to them and timely output processing results. At the same time, because the streaming data has characteristics of mass, continuous, high-speed-arrival, and high timeliness, the processing of streaming data is often based on the time window to delimit the boundaries of the data currently being processed.

The main business scenarios of real-time data processing can be summarized into the following four categories.

1) “Full volume” subscription and analysis for a single topic. The consumer subscribes the raw data directly from a single topic and analyzes the data using full volume of data. This business scenario is usually suitable for scenarios with high data quality and small data volume. Stream data processing tasks subscribe to the “full volume” of data in a specified topic for analysis and calculation. It can meet the “real-time” requirements;

2) “Complex” subscriptions to data for a single topic. The consumer requires to do a “filter” or an “injection” operation for a given topic. This business scenario is suitable for streaming data analysis and processing tasks based on specified data content. The consumer requires analyzing the data which satisfy some specific domain conditions in the raw streaming data and do not need to subscribe to the full volume of streaming data.

3) Subscription and data association data from multiple topics. This scenario is used due to a single data source data cannot fulfill the requirements of the analysis task. On the contrary, multi-streaming needs to be processed for a data association analysis.
(4) Subscription to real-time streaming data and associate it with offline data for analysis. It is usually used for the scenario of real-time streaming data LEFT JOIN an offline table.

The current streaming data processing technologies can be divided into two types. Firstly, a framework-based stream processing engine, which is typically represented by Spark Streaming, and Storm. This kind of streaming processing engine is characterized by the need for developers to develop logic parts in a specific way for framing. Secondly, a “class-like” or “library-like” streaming processing engine, such as Kafka Stream. This kind of streaming processing engine is characterized by providing concrete classes for developers to call directly, and the operation mode of the whole application is mainly controlled by developers. Therefore, current streaming processing engines can be divided into two categories: (a) computing resources of nodes using large data platforms, which uses “Yarn” as a resource scheduling and management tool, such as Spark Streaming, Flink and Storm; (b) Using message queues as a data bus instead of using resource scheduling and management tools, such as Kafka Stream. Spark Streaming and other streaming data processing engines have comprehensive functions and can better support various business scenarios for real-time processing and analysis of current central streaming data. However, due to the various large data platforms in the data center, there are many kinds of services and large number of tasks, and the computing resources of computing nodes are limited, so it is impossible to use sufficient computing resources. Spark Streaming etc. can effectively support Spark Streaming to perform real-time association and processing of multi-source data, which makes it impossible to guarantee the real-time performance of stream data processing and analysis. In the data bus layer of large data platform, node roles are usually data loading services and message queuing services, so they are used frequently for network and disk, but not for memory, CPU and other computing resources. Kafka Stream is a “class-like tool” for streaming data processing. Its deployment and resource scheduling do not depend on yarn and can effectively utilize computing resources in the data bus. However, the overall function of Kafka Stream is relatively immature. It can only support a limited number of scenarios. E.g., it cannot support custom associated data sources, and cannot support association analysis of streaming data in topics and offline data. Therefore, it is impossible for Kafka Stream to fulfill the requirements of streaming data analysis and processing in a real scenario.

In order to solve the problem of data re-subscription and multi-source data association computation, and to achieve high-efficient resource utilization, it is urgent to solve the key technology of Kafka-oriented structured data real-time association, and develop a prototype system.

3.1. Real time Association Technology for Kafka-based Multi Streaming Data
Kafka-based multi-stream data real-time association technology refers to the real-time association of two or more topics of streaming data in Kafka. Its characteristic is that the real-time association of multi-stream data can deal with a variety of real-time association scenarios and provide different association techniques and algorithms. The core technologies include the setting of time window, data expiration method and the processing of disordered data.

“Precise Association” is the key problem for two streams in real-time association. When the association operation is processing, in the same time window, the data in the corresponding windows of multiple streams cannot be guaranteed because of the architecture of distributed message middleware (such as Kafka, RecketMQ, etc.). in order to solve this problem, we propose a new data window pattern “Validity Window” (VM) window satisfies a certain threshold. The data is considered valid in the window, and the data that does not satisfy this condition will be divided into two or more streams.

For example, as shown in Figure 2, for real-time correlation computation between two streams, the D3 and D6 data in stream1 do not satisfy the threshold range of the difference between the data timestamp and the window boundary in the same time window, and the D4 data in stream2 do not satisfy the threshold condition, therefore these data do not satisfy the threshold condition when the correlation computation is performed between the two windows. Dataset of D3, D4, D6 do not participate in association calculation.
3.2. Real-time Association Technology between Streaming Data and Offline Data

Real-time association between streaming data and offline data for Kafka refers to the real-time association between “topic data” in Kafka and data in offline storage engine (such as Hive etc.). It is characterized by the ability to adapt to a variety of offline cache engines, the core technologies include the technology of variety of offline data engines adaptation and offline data update in memory.

The key technology of real-time streaming data association with offline data is that the update frequency of offline data is far less than that of real-time stream data. Usually, after data being loaded into memory in batches, almost no changes are made. For this aspect, offline data can be seen in the technically a special case of multi-stream data association. Therefore, the key technology of Kafka-oriented streaming data and offline data real-time association is to achieve a variety of types of data source adaptation and in-memory data cache for update strategy. The implementation logic is shown in Figure 3. For the case of large amount of data, the offline data is redistributed according to the associated “key”, ensuring that the data distributed on each node is the same as the key, and the streaming data subscribed to Kafka is calculated according to the time window; for the case of small amount of offline data, we operate the data as a global cache: distributing the data to each data association calculation node.

3.3. Association Rules Description and Definition

The description and definition of association rules refers to the unified description, parsing and execution management of user association rules. The feature of the rules is that the design of rule management does not depend on business logic, which means the subscription rule management engine is separated from business logic. Its core technologies include rule description language, multiple data source adaptation, etc.
The key technologies of description and definition of association rules include rule description language, rule parsing and conflict handling method. As shown in Figure 4, the rule engine describes the rules by rule description language. The current business rules are expressed by SQL-like language. For example, JOIN is used to implement the association of multi-stream data or streaming data with offline data, where expression is used to intercept and filter the data. Rule parsing is used to describe the rules. After parsing syntax and semantics, a rule tree is generated and stored in the rule memory. The nodes of each rule tree contain information such as user number, rule content, data source type, data processing method definition and output topic. Conflict handling refers to the handling methods of rules when multiple users propose the same rules, or when the content of rules has inclusion relationship. Specific strategies should ensure that business user rules do not affect each other. When the rules defined by the same user conflict, we do the operation of “rules merge”.

3.4. Prototype System Design and Implementation

The design of the prototype system follows the principles of low platform coupling, friendly interface, user-friendly, simple and easy-to-use. It supports plug-in to increase and manage the data source adapted. It also supports the configuration of running parameters. The prototype system can run efficiently on different scales of large data platforms.

4. Conclusion

In this paper, in order to achieve the real-time data association analysis, we solve the key problem of association streaming data, including real-time association technology for Kafka-based multi-streaming data, real-time association technology between streaming data & offline data and association rules description & definition. We construct a data real-time association computing system for multi-tenant, which has characteristics of high availability and high efficiency.

Acknowledgments

This work was financially supported by Young Fund for CNCERT (Grant Number: 2018QN-05).

References

[1] Information on https://issues.apache.org/jira/secure/ReleaseNote.jspa?version=12343014&styleName=Text&projectId=12310843
[2] Information on http://spark.apache.org/streaming/
[3] Information on http://storm.apache.org/
[4] Information on https://www.confluent.io/product/ksql/
[5] Information on http://spark.apache.org/releases/spark-release-2-3-0.html#structured-streaming
[6] Information on https://flink.apache.org/news/2018/05/25/release-1.5.0.html#flink-15---streaming-evolved
[7] Ge Fu, Xinhua Zhang, Chao Li, Study of Message Data Subscription Based on Multi-Application Big Data Analysis, Journal of Netinfo Security. 11(2017) 44-49.