Proc. of Int. Conf. on Current Trends in Eng., Science and Technology, ICCTEST

A Survey on Data Processing in Bigdata

Revathi K M
Asst.Professor, Dept. Of CSE, DBIT,Banglore

***Abstract*—Big data is collection of large data sets which is difficult to handle using traditional processing application. Data is so large that we call that velocity as big data. Challenging task is to handle distributed file system which should be scalable, flexible and reliable. Big data contain massive amount of data set include size beyond ability to handled by commonly used software tools to manage and process data within elapse time. Big data ranges from few terabytes to petabytes. Big data has technologies and techniques to handle hidden data, complex and massive data. Big data has framework to analyse and handle huge amount of data.**

Keywords—Big data, Hadoop, Spark, Storm, Smaze, Flink

I. INTRODUCTION

What is big data?

Big data is large volume of data that include both structured and unstructured data sets. In early days most companies collect daily information and stored them in database. Then data will be used to forecast or analyse data sets. Today, collecting and maintaining huge amount of data is very difficult. Marketers also collect information that people converse about their brands. Sensors will collect huge amount of data for weather forecasting. This information is treasure that can be mined to gleaned insight to products. Big data contain both structured and unstructured data that is difficult to process by traditional database system and software techniques.

Big data analytics is important for huge companies like face book, Google, yahoo etc to process huge amount of data that is shared in social sites. Massive information will be shared so, there is need of big data analytics to process and analysis massive information. These data are different from structured data which is stored in relation database by 5vs (volume, velocity, variant, value, veracity).

The 5Vs are challenging in big data management system.

Volume:

The amount of data is increasing day by day from megabytes, gigabytes, terabytes, petabytes, so on. It is used to employees created data. Massive amount of data is generated by human interaction in social media. Data storage will increase in future years.

Velocity:

Velocity refers to speed of data process. In many organization speeds processing is main task. In social media

Grenze ID: 02.ICCTEST.2017.1.127

and in credit and debit card transaction data has to process in milliseconds.

Variety:
Data sources can be structured or unstructured. Data can be heterogeneous, files can be any format
It can be text, audio, video, log files etc.80% world data is unstructured and to arrange manage and need to analyse these data.

Value:
It is very important v in big data. Because it is keyword for business and IT infrastructure to store huge amount of data in database. Data will be treasure for organisation.

Veracity:
Big data veracity refers to noise and abnormality in data. Is data that is stored and mined meaningful to problem being analyzed. In scoping out your big data strategy you should keep data clean and should process dirty data that is accumulating in the system.

II. PROCESSING PARADIGM OF BIG DATA

Processing paradigm of big data include methods, tools, and techniques that origin from 3V's (velocity, value, variety) along with batch processing, stream processing and hybrid processing). Batch processing deal with volume, stream processing deal with velocity and hybrid processing deal with both volume and velocity.

Batch Processing Systems:

Batch processing helps to handle large set of static dataset and returning result when computation is complete. it will access all data there in system before batch processing starts. This paradigm won't accept any data after computation begin. The main feature of Batch processing is scalability. To achieve high scalability and dealing with huge volume of data. It is concerned with throughput than latency of individual component. Batch processing uses parallel distributed processing framework such as mapreduce. Apache Hadoop uses batch processing framework.

Apache Hadoop

Hadoop was developed by Doug cutting, working in yahoo. Hadoop is open source software which is reliable, scalable and distributed among different clusters. Hadoop can run application on system with thousands nodes. It distributes files among nodes and allow system to work even though the node failure occur. This approach reduces the risk of catastrophic system failure.

MapReduce

Google created MapReduce-for application development on data centres with hundreds of computing node. MapReduce process large scale data records in clusters. This programming model is based on two function which are map() and reduce() function Map function perform task as master node which takes input and divide into smaller sub modules and distributed among slave nodes. Slave node further divide sub modules that again lead to hierarchical tree structure. Slave node process base problem and passes result to master node. The Map Reduce systems arrange all intermediate nodes together and then send to reduce function for producing final output. Reduce function collect all intermediate node result and combines and put together to form output.

III. STREAM PROCESSING SYSTEMS

The main aim of stream processing paradigm is to deal with velocity of big data. In stream processing it will process data with low latency. It also follows same principles of batch processing such as parallelism and distribution. Stream processing analyse small set of data that are stored in memory to achieve low latency. Real time processing will process infinite sequence of small batch processing where the information is stored in memory instead of disk. Data is processed concurrently and continuously and record by record fashion. Example for real time processing is to define current trend in twitter. Stream processing can handle huge amount of data but only one process at a time.
The datasets are considered as unbounded in stream processing. This has important implication.

- The total datasets is defined as amount of data that has entered system.

- The working datasets are almost relevant, and limited to one data at a time.
- Processing is based on event based and need to be stopped explicitly. Results will explicitly available and updated continuously as new data arrives.

Apache Storm

Apache Storm is real time computation system. Strom is free and open source distributed system. Storm process unbounded datasets for real time processing same as what hadoop did in batch processing system. Storm is simple and fast. Storm can be used with any programming language. Storm process million of data/second per node. Storm has many use cases such as online machine learning, continuous computation, real time analytics etc. It is scalable, fault tolerant, guarantees data processing.

Apache Samze

Apache samze is unbounded collection of same data type. A stream of data can be read by many consumers in system and message can be added and deleted from the stream. Samze collects processing units logically that act on stream of messages and produce output stream. Apache Kafka is used in samze for distributed message brokering. API is providing by samze for creating and running stream tasks on cluster managed by yarn. In this cluster samze runs Kafka brokers and stream tasks are consumers and Kafka streams are producers.

IV. HYBRID PROCESSING SYSTEM

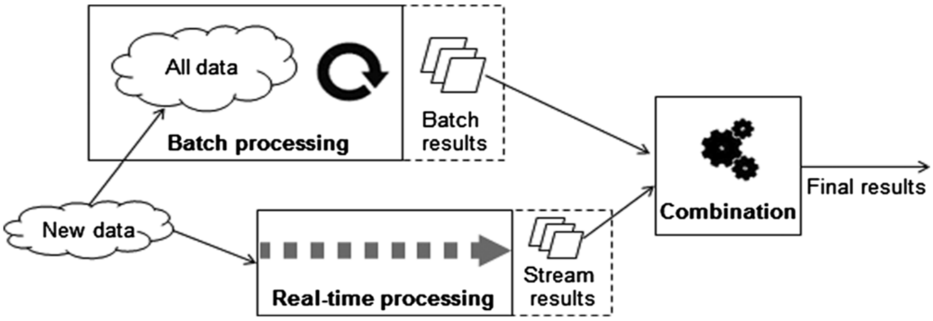

Figure 1. Hybrid processing system

Combination of both batch processing and real time processing is called hybrid processing paradigms. Many applications require hybrid processing. Batch layer manages core dataset, which is stored in distributed files system and is unchangeable ; serving layer (batch results) loads and make visible the batch view in data store so that they can be queried; Serving layer (real time processing)- deal with new data that require low latency. To obtain complete result both batch and stream processing has to be merged together to get result. New data are duplicated and send to both batch layers and real time layers. Batch layer process whole dataset in loop and it takes long time to finish, so new information may arrive during the process and such information is not considered by batch layer. To compensate this delay, real time processing data that have not been analysed by batch layer. Each layer stores partial; result in database that is combined by combination layer to obtain final updated results. Some processing framework handles both batch and real time processing.

Apache Spark

Apache spark is combination batch processing framework with stream processing capabilities. It uses many principles of Hadoop MapReduce engine; it focuses on speeding up batch processing workloads with full in-memory computation and processing optimization. Spark can be deployed as standalone or can use in Hadoop as alternative to Map Reduce engine.

Apache Flink

Apache Flink is stream processing framework that can also handle batch tasks. Batches are considers to simply be data streams with finite boundaries, and treat batch processing as subset of stream processing. It is called as Lambda architecture. Where streams are used for everything and simplifies model.

TABLE I: COMPARISON BETWEEN APACHE SPARK AND APACHE FLINK

Features	Apache Spark	Apache Flink
Computation Model	Based on operator based computation mode	Based on micro-batch model
Streaming engine	Uses streams for workload	Uses micro batches for workloads
Optimization	Flink comes with optimizer that is independent with actual programming interface	Sparks has to be manually optimized
Latency	Low latency and high throughput	High latency
Performance	High	Low
Fault tolerance	High	Low
Memory management	Provide automatic memory management	Provide configurable memory management
Speed	High	Low

V. CONCLUSION

There are many processing paradigm in big data. In batch processing paradigm,that are not time sensitive, Hadoop is good choice which is less expensive.

For stream only workloads, storm support wide language and can process with low latency, but can deliver duplicates and cannot guarantee ordering with default configuration. Samze integrate with yarn and Kafka flexibility, straightforward replication and state management.

For hybrid workloads, spark provide high speed processing and micro batch processing for streaming. It support integrated library and tooling and flexible integration. Flink provide stream processing along with batch processing support. It is highly optimized, can run task written in other flatform, it also provide low latency processing. The best fit depends on state of data process, how requirement are time-bound, and expectation of result.

REFERENCES

[1] Dhole Poonam B, Gunjal Baisa L, “Survey Paper on Traditional Hadoop and Pipelined Map Reduce” International Journal of Computational Engineering Research||Vol, 03||Issue, 12||

[2] Apache HDFS. Available at http://hadoop.apache.org/hdfs

[3] Apache Hive. Available at http://hive.apache.org

[4] Apache HBase. Available at http://hbase.apache.org

[5] Apache Pig. Available at http://pig.apache.org

[6] Yuri Demchenko “The Big Data Architecture Framework (BDAF)” Outcome of the Brainstorming Session at the University of Amsterdam 17 July 2013

[7] Casado R. The three generations of Big Data processing. In Big Data Spain, 2013