A Survey on Big Data Stream Processing Technological

Kanchana R, Dr. Shashikumar D R

Student of M.Tech., Head Of The Department.
Department of Computer Science and Engineering ,
Cambridge Institute of Technology, Bangalore, Karnataka

***Abstract*—The paper reviews on a data stream processing technological in the big data area. We cover the primary elements behind data streams processing Framework. In this survey, we reference a Big Data Processing with Apache Spark,Apache Flink and so on. Our aim is to provide a fast introduction and survey of the technical solutions for big data streams processing.**

***Index Terms*— Data stream, Apache Spark, Streams processing, Big Data, Apache Flink.**

I. INTRODUCTION

The Big Data is an important role in galactic organization. Data is exploding rapidly in different areas of of population growth and technology developments. Cost-effective methods are required to manage the Big Data Analysis. In order to procedure Spark helps to simplify the challenging and compute-intensive task of processing flooding volumes of real-time, both structured and unstructured, seamlessly integrating relevant complex capabilities such as machine learning and graph algorithms. Spark bring forward Big Data processing to the masses.

Different data streams could have own features.Processing frameworks compute over the data in the system,by reading from non-volatile keeping Computing over data is the process of extracting information and insight from large quantities of individual data points. In the same time, the data stream for sensors depends on sampling and so, existing a sample of the entire population. Sometimes, data streams could be buzzing. Spatial and temporal attributes could dramatic work important role in data streams processing. In some cases we have to pay attention the limited resources for data streams processing. Real-time data streams processing will have personal complexness.

II. BIG DATA STREAMS

The demand for stream processing is increasing. Immense amounts of data have to be processed fast from a rapidly growing set of disparate data sources. This pushes the limits of traditional data processing infrastructures. These stream-based applications include trading, social networks, Internet of things, system monitoring, and many other examples.

A number of powerful, easy-to-use open source platforms have emerged to address this. But the same problem can be solved differently, various but sometimes overlapping use-cases can be targeted or different vocabularies for similar concepts can be used. This may lead to confusion, longer development time or costly wrong decisions.

In this section, we discuss some technological solutions for data streams processing.

A. Spark Architecture

Spark can be on a distributed computing framework like Mesons or YARN. Figure 1 below shows the components of Spark architecture model.

Data Storage in Spark uses HDFS file system for data storage goal. It works with whatever Hadoop matched data source including HDFS, Hbase etc.

API in Spark furnish the application developers to make up Spark based applications exploite a standard API interface. Spark furnish API for Java, and Python programming languages.

Resource Administration in Spark can be deployed as a Stand-alone server or it can be along a distributed computing framework like YARN.

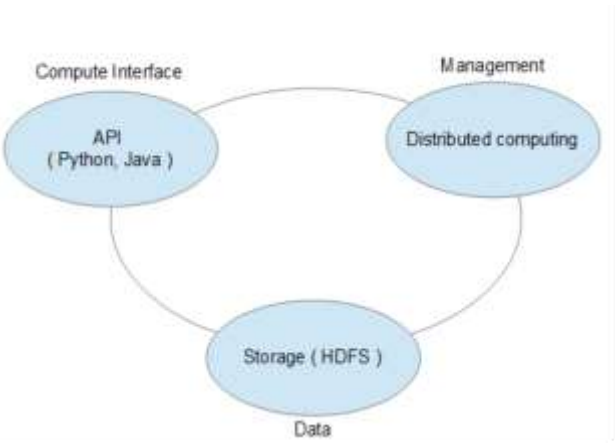

Figure 1. Spark Architecture

B. Spark Streaming

Spark Streaming [2] is an enlargement of the core Spark API [5] that enables scalable, high-throughput, fault-tolerant stream processing of real-time data streams. Data can be absorb from many sources like Kafka, Flume, Twitter, ZeroMQ, Kinesis and can be computerised using complex algorithms expressed with high-level functions like map, reduce, and window (Figure 2).

Figure 2. Spark Streaming

C. *IBM InfoSphere Streams*

IBM InfoSphere Streams [9] is an progressive analytic platform that allows user-developed applications to rapidly ingest, analyze and variable quantity information as it arrives from thousands of real-time sources. The resolution can handle very advanced data throughput rates, up to millions of messages per second .

D. Apache Flink

Apache Flink is an Apache project for dealing with Big Data processing. Although it looks similar to Apache Spark, there are a lot of variation in both their architecture and ideas. The defining trademark of Apache Flink is the power to process the streaming data in real time. Apache Spark is well idea out to be the mastermind in real-time processing with proven capabilities, but its micro-batching architecture assist a Adjacent to Real Time premiss — Apache Flink is plainly actual time.

The kernel of Apache Flink is the Runtime as shown in the architecture diagram beneath. We can also tell it is the Core of Flink which is a distributed streaming dataflow engine that furnish fault tolerant information distribution and communication. The streaming dataflow engine interprets all program as a dataflow graph(Figure 3).

On the upmost of the Center, we have DataStream API for Stream processing and DataSet API for batch processing. There are likewise specific API and Libraries complete the DatasStream and DataSet API's delineate beneath:

Table API enables the custom of SQL queries over the data. They are be well embedded on both the DataStream and DataSets API's and influence the usage of relational operators like selection, aggregations, and joins.

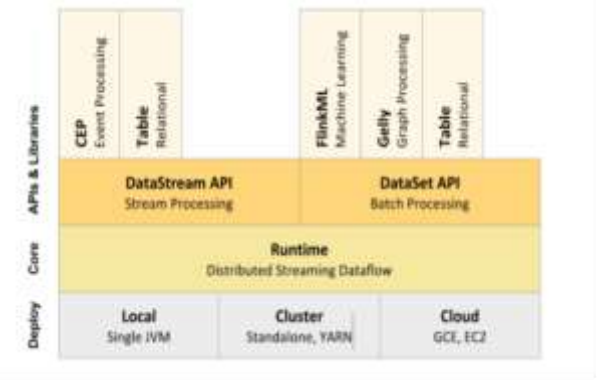

Figure 3. Apache Flink Architecture

Flink ML can be put-upon for performing machine learning tasks over the DataSet API. It enables users to write ML pipelines which make it simple to handle the machine learning workflow. ML pipelines bind the different steps of an ML flow collectively making it efficient to prepare and deploy the models in a manufacture environment.

Gelly for chart processing. It furnish set of operators to create and modify graphs. A chart is described by a DataSet of edges and DataSet of vertices. Dynamite is only accessible for DataSet API and can only be used for batch processing.

Flink CEP is the complex case processing library for Flink. It allows you to quickly detect tangled event patterns in a flow of endless data. Flink CEP is single available for stream processing over DataStream API.

E. Apache Samza

Apache Spark has as its architectural foundation the resilient distributed dataset (RDD), a read-only multi set of data items distributed over a cluster of machines, that is maintained in fault tolerance way.

III. SOCIAL MEDIA CONCEPTS

Nerve-wracking to understand and read the dynamics of an Big Data Processing with Apache Spark has traditionally been a challenging undertaking, often requiring long hours of watching and consultation. This process has been extremely easy thanks to the raise of new methodologies of data recollection and representation, induced by the dissemination of digital technologies. Mainly interesting is the opportunity offered by the collection, organization and interpretation of data coming from social networks, passively provided by users. Therefore, a large number of one-on-one can be studied, without directly involving them in the research: whether the user expresses no restrictions in terms of privacy rights, this data can be easily obtained, deepened and visualized towards proper maps. There are clear advantages in the employ of so-obtained information, primarily because they are automatically produced, free and continuously updated.

Latterly, with the high speed development of the social networks so much as Twitter ,Facebook and Weibo , many researchers have published their work of using the data from social networks including special events for targeted advertising , marketing , localization of natural disasters , and predicting sentiment of investors investigated the real-time nature of Twitter and Facebook, put special attention to event detection.The social media data remarkably has high value In the traffic surveillance system, the social media data can furnish the real time condition of the road network.

On the other hands, the huge volume of social data brings the challenge for mining the value from the social media data .The social media device is with accelerated data in/out. The velocity of collecting social media data is quicker than that of processing and analyzing them. The high velocity of social media devices brings the big challenges for processing and analyzing social media data.

Table 1: Comparative Study of Big Data Streaming Processing Technology Development.

S.No	Research Paper	Focus	Limitations
1	Abdul Ghaffar et.al., [1]	Express the concept of Big Data Analysis from some sample big data source, such as Twitter twits, using the industries emerging tool, known as Spark by Apache.	Analyzed data representation is poor,need to have powerful data representation tool to provide powerful reporting.
2	Abhishek Devarakonda et.al., [3]	Introduction to Apache Spark for users of different experience levels, with a specific focus on the Spark SQL, ML, and SparkR modules.	It currently lags a little bit in the ease of pre-processing data as well as data visualization, but is improving on these capabilities through the use of Spark R and extensions like Plotly.
3	Fatos Xhafa et.al., [4]	In this paper we have presented and evaluated the Yahoo!S4 architecture for real time processing of Big Data Streams. Yahoo!S4 is an alternative to batch mode processing –supported by MapReduce framework– for Big Data Stream processing in real time. The study was conducted in a Cluster environment .	Issues with heteregoneity of computing znodes in the cluster that required adjusting the parameters to improve the busy vs iddle time of the znodes.
4	Yong-Ju Lee*et.al., [8]	The proposed data channel management provided parallel task execution from splitting jobs, and it is useful to process big data even when they are overloaded.	Limitation with the scalability and efficiency for parallel task in processing big data.
5	Fatos Xhafa et.al., [6]	The study was conducted in a Cluster environment which contains heterogeneous nodes. Several frameworks have been proposed for Big Data Streams processing such as Yahoo!S4, TwitterStorm, Spark and Samza.	We would like to address the tuning through self-adapting mechanisms in a more general setting.
6	Shahin Vakilinia et.al., [7]	The proposed algorithm is able to capture the characteristics of the parallel computing, and the experimental result shows accuracy and could be applied in all kinds of stream processing topology structures.	We would like to find the method to help the developers set the parallelism hint in an optimal way to provide the best performance in using Storm in processing real-time stream.

V. ACKNOWLEDGMENT

To quantity up, Spark helps to simplify the challenging and compute-intensive task of processing high mass of real-time both organized and unorganized, rough segregation under consideration tangled capabilities such as machine learning and graph algorithms. In this short paper, we furnish an introduction for stream processing in a big data area. We are planning to furnish a more ankle-deep analysis for the above-mentioned systems in the approaching papers.

REFERENCES

[1] Abdul Ghaffar Shoro & Tariq Rahim Soomro "Big Data Analysis: Ap Spark Perspective "Volume 15 Issue 1 Version 1.0 Year 2015.

[2] SparkStreaminghttp://spark.apache.org/docs/latest/streamingprogramming-guide.html Retrieved: Jul, 2015.

[3] Abhishek Devarakonda, Xiurong Lin, Ryan Borowicz, Jayanti Trivedi MSBA6330 – Gold Cohort -Section 002 Team 5 December 14, 2016.

[4] Fatos Xhafa, Victor Naranjo, Santi Caballe "ProcessingandAnalyticsofBigDataStreamswithYahoo! S4",2015 IEEE 29th International Conference on Advanced Information Networking and Applications.

[5] Shoro, A. G., & Soomro, T. R. (2015). Big Data Analysis: Apache Spark Perspective. Global Journal of Computer Science and Technology, 15(1).

[6] FatosXhafa,VictorNaranjo,LeonardBarolli,TakizawaHose ,"OnStreamingConsistencyofBigDataStreamProcessinginHeterogenousClutsers",2015 18th International Conference on Network-Based Information Systems.

[7] Shahin Vakilinia, Xinyao Zhang and Dongyu Qiu "Analysis and Optimization of Big-Data Stream Processing", in 2016 IEEE.

[8] Yong-Ju Lee*, Myungcheol Lee*, Mi-Young Lee*, Sung Jin Hur*, Okgee Min** ,"Design of a Scalable Data Stream Channel for Big Data Processing "in ICACT2015.

[9] Ballard, C., Brandt, O., Devaraju, B., Farrell, D., Foster, K., Howard, C., ... & Uleman, R. (2014).