Intraday- Big data Analytic for Day Trading

Dr. B Shadaksharappa (HOD, Dept. of CSE, SSCE), hod.cse@sairamce.edu.in
Prof. M Sheela Devi (Dept. of CSE, SSCE), sheela.cse@sairamce.edu.in

Abhinandan Mazumdar
Dept of CSE
Sri Sairam College of Engineering

A Kodeeswaran
Dept of CSE
Sri Sairam College of Engineering

Avinash V
Dept of CSE
Sri Sairam College of Engineering

Bhavana M
Dept of CSE
Sri Sairam College of Engineering

Abstract: ***Big data is a term used to refer to data sets that are too large or complex for traditional data-processing application software to adequately deal with. Data with many cases offer greater statistical power, while data with higher complexity may lead to a higher false discovery rate. An analytics engine for big data processing is very much required as the algorithm used to manipulate the large amount of data. In our project, Big Data Analytics and Real-time Streaming has been used to give the day traders an experience of receiving all the required trading tweets in a single website. The website will post the latest trading news (Corporate announcements to start with) by analyzing the tweets posted by various twitter forums. The server component will analyze and filter only that news which will make a difference in the trading on the market. The news posted will have a time stamp in descending order showing the time at which the news was published. The day traders can achieve more profit by getting the information they need as soon as the new breaks.***

KEYWORDS: **Big Data Analytics, Real-time streaming, trading.**

I. Introduction

Big data analytics is the often complex process of examining large and varied data sets -- or big data -- to uncover information including hidden patterns, unknown correlations, market trends and customer preferences that can help organizations make informed business decisions. Driven by specialized analytics systems and software, as well as high-powered computing systems, big data analytics offers various business benefits, including new revenue opportunities, more effective marketing, better customer service, improved operational efficiency and competitive advantages over rivals. Separately, the Hadoop distributed processing framework was launched as an Apache open source project in 2006, planting the seeds for a clustered platform built on top of commodity hardware and geared to run big data applications. By 2011, big data analytics began to take a firm hold in organizations and the public eye, along with Hadoop and various related big data technologies that had sprung up around it. An example of where this is useful for financial services organizations is with regard to real-time risk assessment. In this case, the full life cycle of a transaction needs to be considered, which includes past (the trade was agreed to and executed) and present (post-trade events like amendments, transfers and terminations). A real-time alert of a trade event, although important, is more valuable in the context of the larger historical picture of the entire set of portfolios. To do this at scale is crucial because the domino effect of any series of real-time events on a larger system of trading strategies could be the difference between success or a massive loss that triggers regulatory scrutiny and reputational risk. Real time streaming dashboard can be very useful for a day trader to receive his information in a single place.

Fig: Front End Dashboard

II. SYSTEM ARCHITECTURE

The architectural design of a system emphasizes the design of the system architecture that describes the structure, behavior and more views of that system and analysis. In our project the system architecture is of type 2-tier architecture. The two tiers are server tier and the app tier. In the server tier, the twitter data will be collected using a python code and stored in real-time distributed streaming platform APACHE Kafka server. Topics store the tweets and keep it until they push it to the next component APACHE Spark which is analytics engine which provides high processing power and in-built algorithms for big data analytics. The algorithm we designed in spark will do the rest of the job. The tweets will be analyzed and then sent in the order to the next topic in APACHE Kafka server. Then the collected tweets will be published on the website handle which is connected to it. The architecture is very simple but the algorithm's logic is very critical and difficult to create and analyze. In the app tier, a script will be running all the time to update the real-time data which has been analyzed and sent by the Kafka server. AJAX request will be used so that only the part where updating required is only loaded from the server.

.

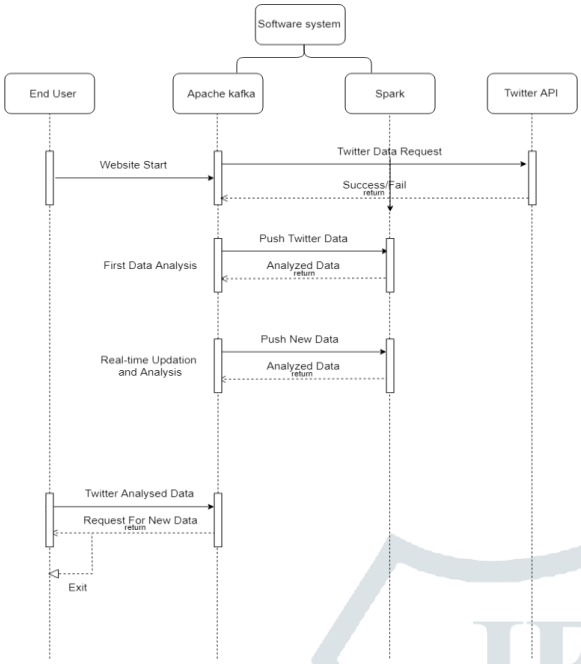

Fig: System Architecture Diagram

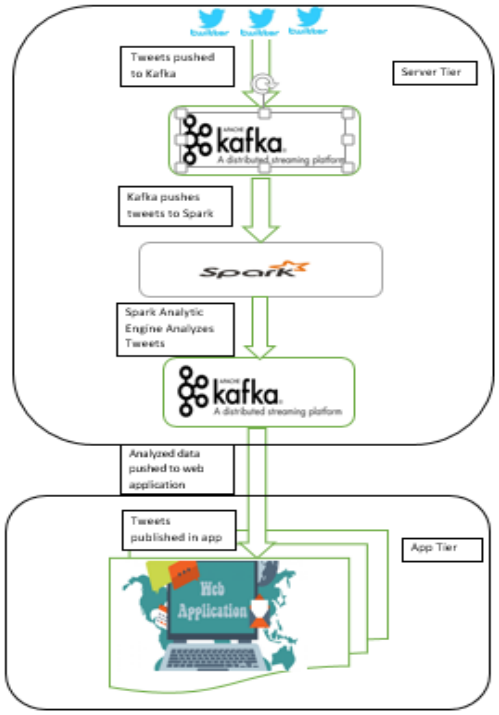

Fig: Data Flow Diagram

III. DATA FLOW DIAGRAM

A Data flow diagram (DFD) is a graphical representation ion of data processing of the flow of the data through an information system, modeling its process aspects. A DFD is often used as preliminary step to create an overview of the system, which can be elaborated later. DFD can also be used for visualization of data processing. A Data shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about timing of the process or information about whether processes will operate in sequence or in parallel.

The two tiers are server tier and the app tier. In the server tier, the twitter data will be collected using a python code and stored in real-time distributed streaming platform APACHE Kafka server. Topics store the tweets and keep it until they push it to the next component APACHE Spark which is analytics engine which provides high processing power and in-built algorithms for big data analytics. The algorithm we designed in spark will do the rest of the job. The tweets will be analyzed and then sent in the order to the next topic in APACHE Kafka server. Then the collected tweets will be published on the website handle which is connected to it. The architecture is very simple but the algorithm's logic is very critical and difficult to create and analyze. In the app tier, a script will be running all the time to update the real-time data which has been analyzed and sent by the Kafka server. AJAX request will be used so that only the part where updating required is only loaded from the server. The data flow model for the project is as follows.

IV. SYSTEM IMPLEMENTATION

Step 1: Apache Kafka: An open source distributed event streaming platform for building real-time data pipelines and stream processing applications

Step 2: Apache Spark: Unified analytics engine for big data processing, with built-in modules for streaming, SQL, machine learning and graph processing

Step 3: News Analytic Engine: The Server component which is used to fetch the data from twitter

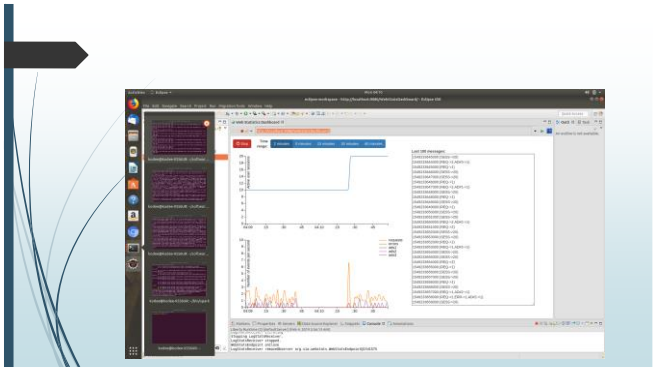

Fig: Server graph and data incoming

V. SOFTWARE ENVIRONMENT

Big data analytics is the often complex process of examining large and varied data sets -- or big data -- to uncover information including hidden patterns, unknown correlations, market trends and customer preferences that can help organizations make informed business decisions.

Driven by specialized analytics systems and software, as well as high-powered computing systems, big data analytics offers various business benefits, including new revenue opportunities, more effective marketing, better customer service, improved operational efficiency and competitive advantages over rivals.

Unstructured and semi-structured data types typically don't fit well in traditional data warehouses that are based on relational databases oriented to structured data sets. Further, data warehouses may not be able to handle the processing demands posed by sets of big data that need to be updated frequently -- or even continually, as in the case of real-time data on stock trading, the online activities of website visitors or the performance of mobile applications.

As a result, many of the organizations that collect process and analyze big data turn to NoSQL databases, as well as Hadoop and its companion tools, including:

- YARN: a cluster management technology and one of the key features in second-generation Hadoop.
- MapReduce: a software framework that allows developers to write programs that process massive amounts of unstructured data in parallel across a distributed cluster of processors or stand-alone computers.
- Spark: an open source, parallel processing framework that enables users to run large-scale data analytics applications across clustered systems.
- HBase: a column-oriented key/value data store built to run on top of the Hadoop Distributed File System (HDFS).
- Hive: an open source data warehouse system for querying and analyzing large data sets stored in Hadoop files.
- Kafka: a distributed publish/subscribe messaging system designed to replace traditional message brokers.

Fig: Software Architecture

VI. TESTING

The use of testing the system is to identify the errors. Testing is regarded as the process of trying to identify every conceivable false or weakness present in the work product. It will provide a way for checking the functionality of components, assemblies, sub assemblies, and a finished product.

TYPES OF TESTS

The Unit testing:

The Unit testing will involve the designing of test cases which validate that the internal program logic is properly functioning, and the inputs of that program will produce the valid outputs. All of the decision branches and the flow of internal code must be validated. It is considered as the testing of individual software units for the application.

The Integration testing:

The Integration tests were designed for testing the integrated software components in order to determine when they actually run as one program. Testing is considered as an event driven and it is more concerned by the basic outcome of fields or screens.

The Functional test:

The Functional tests will provide a systematic demonstration that the functions which are tested were available as described by the business and the technical requirements, user manuals and the system documentation.

The System Test:

The System testing will ensure that the entire integrated software system will meet the requirements. A configuration is tested to ensure predictable and known results. The example of system testing is the configuration of oriented system integration test. The System testing will be based on the process flows and descriptions, emphasizing the pre-driven process links and the integration points.

VII. PERFORMANCE EVALUATION &VALIDATION

Test Results

All of the test cases that are mentioned above are passed successfully. No defects were encountered.

The Acceptance Testing:

The User Acceptance Testing is the critical phase in any project and it requires the significant participation from the end user. It will also ensure that the system will meet the functional requirements.

IX. CONCLUSION & FUTURE ENHANCEMENT

The system proposed is a simple website with a very high capability in analyzing data in the backend using powerful software's like Apache Spark and Apache Kafka. The real-time streaming of data will improve the update time and information availability. The single website provides the day trader with easy access to all the trading information from the ocean of twitter and also organized accordingly.

The tweets will be in descending with a time stamp and the tweets which have the ability to create a break in the stock market will be displayed first and the other information will be in the same table with other tweets. The system which we design will contain the ability to update every second or more according to the tweets.

There is no website separately to display all the day trading information in one place, this is the first of its kind with an analytics engine and real-time streaming dashboard.

The future enhancement of the system tends to analyze sentiment of the user which he/she tends to search using machine learning. It will analyze the user's tends to a particular or similar type of shares.

X. REFERENCES

[1] K. Aziz, D. Zaidouni and M. Bellafkih, "Real-time data analysis using Spark and Hadoop," 2018

[2] 4th International Conference on Optimization and Applications (ICOA), Mohammedia, 2018, P. Le Noac'h, A. Costan and L. Bougé, "A performance evaluation of Apache Kafka in support of big data streaming applications," 2017 IEEE International Conference on Big Data (Big Data), Boston

[3] B. Yadranjiaghdam, S. Yasrobi and N. Tabrizi, "Developing a Real-Time Data Analytics Framework for twitter Streaming Data," 2017 IEEE International Congress on Big Data (BigData Congress), Honolulu, HI, 2017

[4] G.M. D'silva, A. Khan, Gaurav and S. Bari, "Real-time processing of IoT events with historic data using recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, 2017

[5] Monu Kumar and Dr. Anju Balu, "Analyzing Twitter Sentiment through Big Data", 2016