Real-time Data Streaming and Analysis

1st Sai Charishma Kurmala
Department of Applied Data Science
San Jose State University
San Jose, United States
saicharishma.kurmala@sjsu.edu

2nd Fnu Maria Poulose
Department of Applied Data Science
San Jose State University
San Jose, United States
mariapoulose@sjsu.edu

3rd Divya Nalam
Department of Applied Data Science
San Jose State University
San Jose, United States
divya.nalam@sjsu.edu

4th Harika Boyina
Department of Applied Data Science
San Jose State University
San Jose, United States
harika.boyina@sjsu.edu

5th Madhulika Dutta
Department of Applied Data Science
San Jose State University
San Jose, United States
madhulika.dutta@sjsu.edu

***Abstract*—In the global economic landscape, stock markets serve as crucial platforms for companies to secure the capital necessary for their daily operations. The significance of conducting research in the stock market domain becomes evident given the inherently dynamic and unpredictable nature of these markets, shaped by factors such as media coverage, investor sentiments, corporate strategies, and political considerations. The substantial daily engagement of numerous investors underscores the considerable demand for accurate predictions of stock prices. This project aims to contribute to this field by developing a real-time dashboard that captures and visualizes streaming data, enhancing the accessibility and usability of critical information for professionals, investment communities, and enthusiasts. Our primary focus is on constructing a scalable and efficient data pipeline designed to seamlessly handle the ingestion, processing, analysis, and visualization of substantial volumes of real-time streaming data. The methodology involves harnessing big data technologies to deliver real-time insights, empowering businesses to make prompt and informed data-driven decisions. This entails the development of a robust data ingestion system, the implementation of streaming analytics algorithms, and the creation of interactive dashboards tailored for visualization purposes**

***Index Terms*—Startup, Success prediction, Random Forest, Logistic Regression**

I. Introduction

Stock data stands as a reservoir of invaluable information crucial for diverse stakeholders in the financial system. Traders and investors leverage historical stock prices, volume, and other variables for technical analysis, identifying patterns and trends that inform their buying and selling decisions. Meanwhile, fundamental analysis, rooted in financial statements and ratios derived from stock data, enables an assessment of a company's health and future growth potential. Real-time stock data holds particular significance for algorithmic trading systems and day traders, facilitating swift decision-making aligned with market conditions.

The landscape of stock data utilization has undergone transformative shifts with the application of advanced analytics and big data. Artificial intelligence and machine learning algorithms meticulously scrutinize extensive datasets, uncovering hidden patterns that illuminate market behavior and forecast future trends. This transformative journey is further propelled by the rising prominence of alternative data sources, including social media sentiment and satellite imagery, providing additional dimensions for comprehensive research.

In essence, stock data emerges as an indispensable linchpin in financial markets, steering decision-making processes, fortifying risk management practices, and advancing our comprehension of the intricate dynamics that shape the global economy.

II. Significance to real world

In a tangible application to the real world, the profound insights gleaned from our project on developing a real-time dashboard for stock market data hold immense significance for diverse stakeholders. Traders and investors stand to benefit from more informed decision-making processes, leveraging the real-time analytics to adapt swiftly to market dynamics. Financial professionals gain a valuable tool for risk management, allowing them to navigate the unpredictable nature of stock markets with greater precision. Moreover, businesses and policymakers can utilize the findings to bolster economic strategies, responding proactively to market trends and investor sentiments. The real-world impact extends beyond the financial sector, contributing to a broader understanding of economic dynamics and fostering a more resilient and adaptive global economy.

III. Project Motivation

Our project, "Real-time Data Streaming and Analysis in Stock Markets," is motivated by a combination of technological advancement and economic necessity. Here are some key

points that highlight the motivation behind our project: Need for Real-time Insights in Financial Markets: Stock markets are dynamic and fast-paced. Decisions need to be made quickly, often within fractions of a second. By providing real-time data analysis, our project empowers investors and financial analysts to make more informed and timely decisions. Advancements in Big Data Technologies: The availability of advanced big data technologies, like AWS Lambda for serverless computing, allows for more scalable and efficient data processing. Our project demonstrates the practical application of these technologies in a high-stakes environment like the stock market. Empowering Data-Driven Decision Making: In a domain where decisions are based on data, our project enhances the ability of businesses and investors to make data-driven decisions rapidly and accurately. Educational Value in Big Data Learning: As a part of our big data course, this project not only serves a practical purpose in the financial world but also contributes to our educational journey, helping us understand and apply big data concepts in real-world scenarios.

In summary, our project stands at the intersection of finance, technology, and education, addressing key challenges in the stock market through innovative use of big data technologies, and offering valuable insights for various stakeholders in the financial ecosystem.

IV. Literature survey

Big data analytics is currently gaining enormous popularity across all industries. Large volumes of data are generated and processed on a daily basis with the use of big data analytics tools and techniques. Scala is the language utilized here, and Apache Spark is the technology. Therefore, the study conducted in "Propositional Aspect between Apache Spark and Hadoop Map-Reduce for Stock Market Data" paper is based on research conducted utilizing the Apache Spark technology on stock market data. Here, the influence of COVID-19 is analyzed using nifty-50 data. Since COVID-19 has been shown to have an impact on nearly everything in the world, the goal is to examine how it has affected the stock market. After then, a comparison is made between the methods employed to examine the enormous amount of stock exchange data mentioned in this paper. An examination of several key parameters has been done in this research to compare Apache Spark and Hadoop Map-Reduce. That means that when it comes to analyzing stock exchange data, the Apache Spark technique is superior[1].

Social media and Internet of Things (IoT) services are generating enormous amounts of data every second as a result of the web's explosive growth. In addition to being enormous, this data grows swiftly and is challenging to evaluate. Such data cannot be processed in real-time by the majority of existing big data frameworks. Numerous businesses and academics have begun to create new big data frameworks in order to process the data in real time. Real-time data processing has been made possible with the introduction of Apache Spark, Apache Flink, and Apache Storm. Analyzing streaming data has become more effective with the new processing frameworks. One popular area for analyzing large streams of data is stock market analysis. In "Stock Market Analysis from Twitter and News based on Streaming Big Data Infrastructure" paper, they developed a real-time processing system to examine tweets and determine whether they are correlated with the market for stocks. their system's performance and configuration are explained. They obtained 80 percentage separation of increase/decrease of stock value with 77 percentage accuracy of Twitter data classification[2].

For precise prediction and analysis of sizable data sets, big data analytics are primarily utilized in a variety of industries. They make important information that would otherwise be hidden from view accessible from massive data sets. The robust Cloudera-Hadoop based data pipeline approach presented in this paper can handle analyses of any size and type of data. In this case, real-time Yahoo Finance data is used to predict daily gains by analyzing a selection of US stocks. Using the Spark machine learning module, stocks with high daily gains are predicted by dividing the daily gain data of the US stock market's stocks into training and test sets in this "Stocks Analysis and Prediction Using Big Data Analytics". The Hadoop big-data framework is used to handle large data sets through distributed storage and processing[3].

The majority of investors are drawn to the stock market because of its high payout. Nonetheless, a wide range of factors influence how much a stock trade fluctuates in value. Investors can employ a variety of measuring instruments to lower their investment risk. Predicting the price of stocks is a crucial subject in the financial market. Forecasting financial time series is a common use of machine learning techniques. According to earlier studies, sophisticated forecasting techniques are capable of precisely predicting changes in trading prices in financial markets. This study aims to send risk notifications based on various trading volume levels and analyze real-time stock trading volume according to the bursting comprehensive index of trading volume using the real-time stream data processing architecture in the big data Spark framework. Investing can be done at any time of year. The study's of the "Early Warning System in Volume Burst Risk Assessment of Stock with Big Data Platform" paper findings demonstrate that stock trading volume risk rating criteria can be used by investors to boost profits when they wish to engage in high-risk trading[4].

Real-time processing of streaming data is the foundation for the great majority of big data-driven applications and solutions in today's technological contexts. Big data driven applications and solutions are developed in large part through the real-time processing and analytics of data streams. This study defines a massive data processing lifecycle in real time from this point of view. It links the phases of the lifecycle—data ingestion, data storage, stream processing, analytical data store, analysis, and reporting—with the tools, tasks, and frameworks that are currently in use. The paper "Real-Time Processing of Big Data Streams: Lifecycle, Tools, Tasks, and Challenges" also looks into the real-time big data processing solutions, which include Sap Hana, Hbase, Hive, Cassandra, Splunk, Flume, Kafka, Nifi, Storm, Spark Streaming, S4, Flink, and

Samza.The most recent issues surrounding real-time big data processing are also covered, including ”volume, variety, and heterogeneity,” ”data capture and storage,” ”inconsistency and incompleteness,” ”scalability,” ”real-time processing,” ”data visualization,” ”skill requirements,” and ”privacy and security.” This paper could offer significant perspectives on the comprehension of the large data processing lifecycle, associated instruments and duties, and difficulties of real-time processing[5].

Transportation systems depend heavily on the ability to predict flows, such as the movement of bikes, cars, and people, which are made up of the in-and-out traffic at a node and the transitions between other nodes in a spatiotemporal network. But this is a highly difficult subject, influenced by a number of intricate aspects, including the temporal correlation between distinct time intervals, the spatial association between different locations, and external factors (such weather and events). Furthermore, there is mutual effect between the flow at a node, referred to as node flow, and the transitions between nodes, referred to as edge flow. suggested a multitask deep-learning framework that predicts both the node flow and edge flow in a spatio-temporal network simultaneously in order to solve these problems. Their method creates two complex models based on fully convolutional networks to forecast node flow, edge flow. Through the coupling of the latent representations of middle layers, these two models are trained in tandem. Through a gated fusion mechanism, the external element is also included into the framework. They use an embedding component in the edge flow prediction model to handle the sparse transitions between nodes. Their approach is assessed using data from Beijing and New York City taxicabs. Their technique outperforms 11 baselines, including ConvLSTM, CNN, and Markov Random Field, as demonstrated by experimental findings [6].

Consumers are constantly eager to receive responses from analytics systems immediately. The value is lost if it takes longer than a few milliseconds to gain insight. Applications that depend on data from Twitter feeds, sensors, stock markets, or fraud detection cannot afford to wait. This frequently entails reviewing incoming data before it is even entered into the record database. The task becomes significantly more difficult when zero tolerance for data loss is included. In a real-time big data scenario, streaming analysis allows us to spot trends and act upon them as soon as data starts to come in, as opposed to waiting for data to be gathered in its entirety at a lengthy periodic interval. Streaming systems adjust themselves when data are non-stationary and patterns change over time. This article describes methods for processing and storing data in close to real-time for the purpose of analyzing data streams related to the Cassandra NoSQL database. It offers insight into how to best utilize Cassandra in a multi-data center configuration for almost real-time responses. The traditional trade-off between high accuracy and low latency is formulated. The theoretical assertions are supported by multiple comprehensive experimental analyses in the Apache and Datastax Cassandra distributions [7].

V. PROJECT OVERVIEW AND ARCHITECTURE

A. Methodology

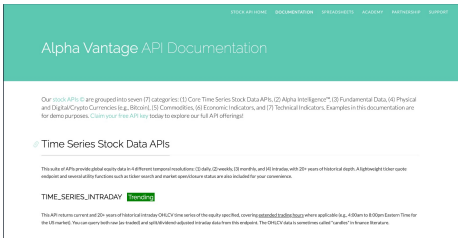

Fig. 1. Dataset

For this project, we consider Alpha Vantage API as our data producer.

The Alpha Vantage API offers different types of stock market data, organized into seven categories. Among these, the ”Time Series Stock Data APIs” is particularly important for those interested in stock market trends.

The ”TIME-SERIES-INTRADAY” part of this API provides detailed stock information that is updated within the day, called intraday data. This means it can give you the stock data not just at the end of each trading day, but also at several points during the day. This data includes:

Open: The price at which the stock started trading at the beginning of the day or a specific time interval within the day. High: The highest price of the stock during the day or a specific time interval. Low: The lowest price of the stock during that day or time interval. Close: The price at which the stock ended trading at the end of the day or time interval. Volume: The number of shares traded during the day or specific time interval.

The stock symbol we considered is ’IBM’. The rate set is 1 minute.

In Amazon web services, we have created a lambda function named project-stream-data function to ingest data from Alpha Vantage API and Alpha Vantage is a well-known source of financial data, and this Lambda function is made to ingest and store the stock data for later processing and analysis on AWS. Attached required IAM roles i.e kinesis full access, lambda basic to the lambda function. A layer has been implemented to improve the maintainability and efficiency of the Lambda function that is in charge of ingesting data from Alpha Vantage. By including the libraries and dependencies needed to interface with Alpha Vantage APIs, this layer helps to promote modular code architecture and minimizes the size of the deployment package. The layer guarantees that the Lambda function stays succinct and concentrated on its essential logic while utilizing external dependencies that are kept in the layer with ease. It also makes the development and deployment process more efficient.

Created a kinesis data stream named stock-stream-data. The Alpha Vantage API is used by this AWS Lambda function to retrieve intraday IBM stock data. It then processes the data and broadcasts it into stock-stream-data(kinesis data stream).

It manages problems in data retrieval and streaming, leverages environment variables for the API key and stream name, and returns pertinent results. In order to maintain the Kinesis stream updated with the most recent stock data, the function is intended to be executed. With the use of an eventbridge CloudWatch, this AWS Lambda function is set up to execute every minute. Using the CloudWatch event scheduling feature, the Lambda function is automatically triggered once per minute to begin the process of retrieving IBM's intraday stock data in real time from the Alpha Vantage API. Next, formatted data is sent into an Amazon Kinesis Data Stream without any interruption. This system is ideal for keeping IBM's Kinesis stream current with the most recent one-minute interval stock information since its event-driven architecture guarantees frequent and timely updates. A thorough view of the lambda function, layer, cloudwatch events, and the code we used to link it to Alpha Vantage can be seen in the fig 1 below.

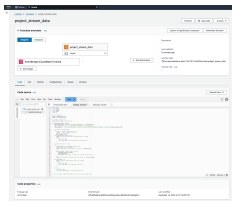

Fig. 2. Lambda function to ingest data from Alpha Vantage API

Effectively broadcasting data to the "stock-stream-data" Kinesis Data Stream after it has been processed by the Lambda function which is shown in the below screenshot. Comprehensive metrics including "Get Records Success," "Get Records Sum," "Get Records Latency," "Get Records Iterator Age," "Incoming Data - Sum," and "Put Record - Sum" are revealed through screen captures. A concise summary of the Kinesis stream's performance is given by these visualizations, which also offer insightful information on the operational dynamics of the workflow for data transportation and processing.

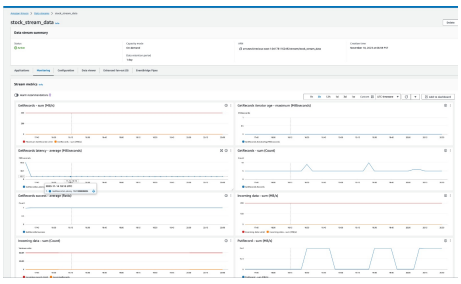

Fig. 3. Raw data ingetsion - monitoring (1st kinesis data stream)

An extra Lambda function called 'process-stream-data,' has been developed specifically for preparing data that has been extracted from a Kinesis Data Stream. Prior to the data being sent to the OpenSearch endpoint, this function is essential in cleaning up the data.

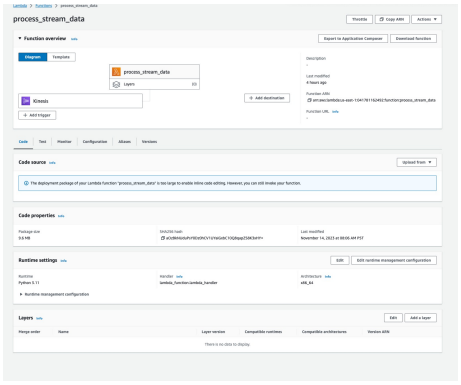

Fig. 4. Lambda function using for formatting data

To store processed data, I have created another new Kinesis Data Stream called "stock-sma-stream." The metrics 'GetRecords,' 'GetRecords Iterator Age,' 'GetRecords Latency,' and 'GetRecords - Sum (Count)' are visible and indicate the dynamic activity that occurs within the 'stock-sma-stream' Kinesis stream. This is illustrated in the screenshot that has been presented.

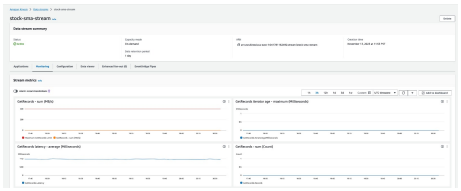

Fig. 5. Processed data stream - monitoring

The processed data stream was successfully transmitted from the Lambda function(process-stream-data) to the OpenSearch Service, as confirmed by the CloudWatch logs. This shows how well the data processing pipeline functions, and below screenshot tells how CloudWatch logs provide real-time monitoring.

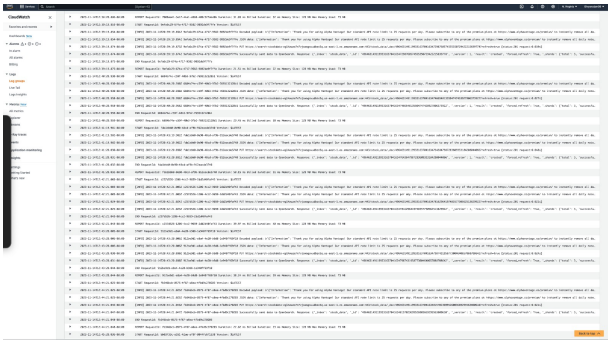

Fig. 6. Cloudwatch logs showing processed data stream sen successfully to OpenSearch Service

It is confirmed that the data has been successfully ingested and is available for exploration within the OpenSearch environment by the fact that items are clearly visible under the 'Discover' tab at the OpenSearch service endpoint.

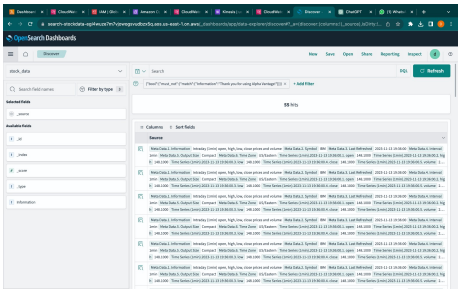

Fig. 7. Data received to opensearch service endpoint and can see entries under discover tab

In the OpenSearch environment, a filter has been implemented to handle warnings produced by the API. Final

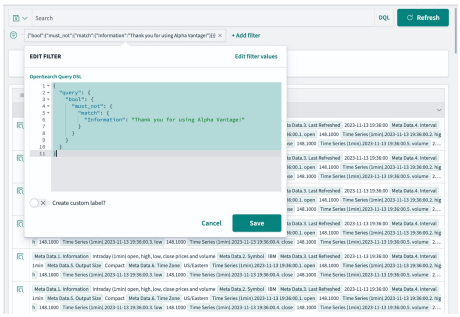

Fig. 8. Filter attached to deal with warnings from API

Dashboard of our project is shown in the fig 8 below

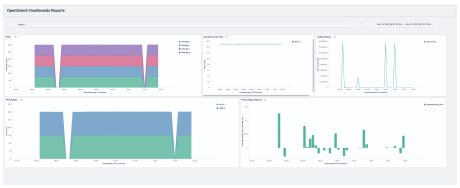

Fig. 9. Final Dashboard

B. Architecture and Tools

Below is the workflow of our project.

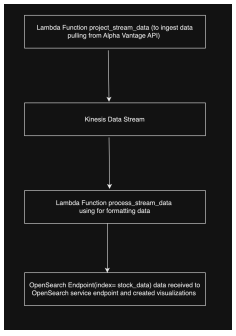

Fig. 10. Project Workflow

We use the following industry standards tools to complete this project:

Trello - Agile project life-cycle management , we use Trello to manage the project life cycle. We usually have meetings every week to work on our project.

To ensure clarity and correctness in our report, we utilized Grammarly, a powerful language-enhancement tool. Its comprehensive grammar, spelling, and punctuation checks helped us maintain a high standard of academic writing.

Microsoft Office Suite is used for all project documentation, presentation, and data exploration.

Grammarly - for proof-reading documentation's grammar. This is shown in below screenshot below.

Python (programming language) is used for data processing.

Overleaf is a cloud-based document preparation platform that enables real-time document creation, editing, and collaboration.

GitHub - version control for our project.

Draw.io is a diagram visualization software.

Our project employs a robust, serverless architecture designed to process and analyze real-time stock market data. The architecture is composed of several AWS services that work in tandem to provide a seamless data flow from ingestion to visualization.

OpenSearch(Data Visualization): The final component of our architecture involves creating visualizations from the indexed data in OpenSearch. Utilizing the service's built-in visualization tools, we construct interactive dashboards that present the real-time analysis of stock data. These visualizations are pivotal in presenting the data in an accessible and comprehensible manner, facilitating quick and informed decision-making for users. OpenSearch provides the capabilities for full-text search and data analytics along with built-in visualization tools. Here, the data is indexed, making it searchable and ready for further analysis.

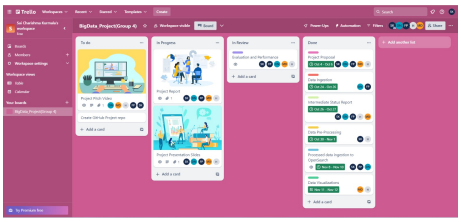

Fig. 11. Trello

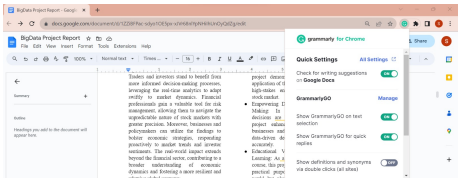

Fig. 12. Grammarly

VI. LESSONS LEARNT

We gained profound insights and developed a comprehensive understanding of various cloud-based services and tools during the course of our project, which focused on harnessing real-time stock market data for analysis and visualization. Our project not only made a contribution to the field of financial data analysis, but it also provided us with valuable learning opportunities. It provided us with hands-on experience with big data technologies and their applications in real-time environments, which we can apply to future endeavors in data science and analytics.

VII. TEAM WORK

Here we have described team members and roles in below fig 13.

Team members	**Roles**
Sai Charishma Kurmala	Data Analyst, Kinesis Data Stream Specialist
Divya Nalam	Project Lead, Data Streaming Architect
Fnu Maria Poulose	Cloud Engineer, AWS Lambda Developer
Harika Boyina	OpenSearch Dashboard Developer, Visualizer, Apache spark Specialist
Madhulika Dutta	Quality Assurance, Documentation Specialist

Fig. 13. Team Members and Roles

VIII. INNOVATION

Our project, which centered on creating a real-time dashboard for stock market data analysis, pioneered several novel practices and techniques that significantly improved the data analytic workflow. Among these innovations are:

Serverless Data Ingestion and Processing: Our project benefited from a serverless architecture by utilizing AWS Lambda for data ingestion and processing. Because of this novel approach, we were able to automatically scale up or down based on demand, resulting in a more efficient and cost-effective solution.

Real-time Data Streaming with Kinesis: The use of Amazon Kinesis to handle real-time data streams was a game changer. It enabled the processing of large amounts of stock market data with low latency, allowing for immediate analysis and decision-making.

Interactive Data Visualization with OpenSearch Dashboards: A key innovation was the use of AWS OpenSearch (formerly Elasticsearch Service) for data visualization. We used its powerful search and analytics engine to create interactive, real-time visualizations that provide instant insight into stock market trends and movements.

Cloud-based Search and Analytics Engine: Using AWS OpenSearch Service as our search and analytics engine simplified indexing and querying large datasets. This not only improved our search capabilities, but also provided robust analytics features that traditional database systems did not provide.

API Data Retrieval Optimization: The Alpha Vantage API's integration into our serverless functions was optimized for efficiency. Our Lambda functions were designed to handle API rate limits and data normalization, ensuring a consistent data flow.

The ability of our project to simulate a real-time, scalable, and secure analytical environment that is adaptable to the fast-paced nature of stock market fluctuations was, in essence, its innovation. The combination of serverless computing, real-time data streaming, and advanced data visualization practices exemplified a cutting-edge approach to cloud-based big data analytics.

IX. PAIR PROGRAMMING

Because many of us come from diverse backgrounds, we make progress by coding and working in pairs. Members with Python experience help those who aren't, and others help by providing ideas and strategies for working with data. Those who excel at documentation and visualization are assisted by others who offer advice. We have used Github for pair programming as shown in the above fig 14. In Github we have uploaded all our files related to our project.

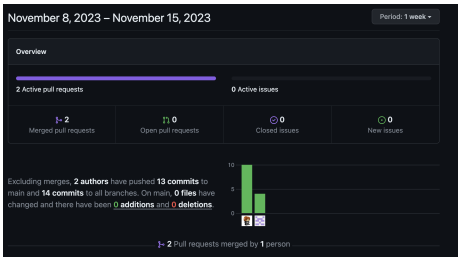

Fig. 14. Pair Programming

X. RELEVANCE TO THE COURSE

Integration of Core Big Data Concepts and Technologies: Our project directly applies key concepts and technologies covered in a Big Data course. It entails dealing with large amounts of data in real time, which is a major challenge in the field of big data. You are utilizing advanced big data technologies and cloud computing by utilizing AWS Lambda, which are essential components of modern big data courses.

Demonstrating Scalability and Real-time Processing: Scalability is an important aspect of big data systems. Our project demonstrates how to create scalable systems capable of processing real-time data streams, which is a necessary skill in the big data domain. This is directly related to the course

goal of comprehending and applying scalable data processing techniques.

Data Ingestion and Streaming Analytics: The project focuses on the ingestion and analysis of streaming data, which is an important component of the big data curriculum. It shows how to effectively capture, process, and analyze data in real time, which is essential in the field of big data.

Visualization and Interpretation of Big Data: Developing interactive dashboards for data visualization in our project reflects an important aspect of big data - making complex data understandable and actionable. This corresponds to course components that emphasize data interpretation and visualization.

Learning about Data-Driven Decision Making: The project demonstrates how big data can be used to inform decision-making processes. This is an important learning outcome for any big data course because it emphasizes the importance of data in guiding business and financial strategies.

Exploring the Impact of Big Data on Different Sectors: Our project's use of big data in the stock market is an excellent example of how big data techniques are revolutionizing various industries. This relevance is likely to be an important component of your course, which aims to provide a comprehensive understanding of the impact of big data across various domains.

XI. TECHNICAL DIFFICULTIES

API Request Limitation: The free API key from Alpha Vantage limited us to only 25 requests per day, posing a significant constraint for continuous data access. To overcome this, a premium subscription is necessary for more extensive and frequent data usage. Complex Integration with AWS Lambda and OpenSearch Service. Pushing processed data from AWS Lambda to OpenSearch Service involved several complex steps: OpenSearch Service Endpoint Setup: Configuring the OpenSearch service endpoint was crucial for a seamless connection between AWS Lambda and OpenSearch. Index Pattern Creation: Creating an index pattern in OpenSearch was essential for efficient data organization and retrieval. Data Formatting: Proper formatting of the Alpha Vantage API data was required to ensure compatibility with OpenSearch. Data Fields and DataTypes Management: Correctly mapping data fields to their appropriate data types in OpenSearch was vital for accurate data representation and analysis.

XII. NOVELTY UNIQUENESS

The innovative use of OpenSearch stands out as a significant contribution to the domain of real-time data streaming and analysis in the stock market. Real-time Analysis and Visualization: The project takes advantage of OpenSearch's real-time analytics and visualization tools to provide a live dashboard of stock market dynamics. This enables an unprecedented level of responsiveness for financial analysts and investors who rely on up-to-the-minute data to make informed decisions. In summary, the novelty of our project lies not only in the technical implementation but also in its pioneering approach to financial data analysis. We have ventured into a relatively unexplored area with OpenSearch, overcome its complexity to build functional and insightful visualizations for IBM data, and in doing so, have set a precedent for future explorations and innovations in this space.

XIII. IMPACT

Our project's driving force was to use the power of big data and real-time analytics to make a tangible impact on the financial sector. We hope to change the way data is used to make critical investment decisions by providing a dashboard for stock market analysis. In summary, our project aims to generate a cascade of benefits ranging from individual investor gains to systemic improvements in financial markets. The capabilities we developed for real-time data streaming and analysis have the potential to influence and refine financial decisions made at all levels, from individual investors to large financial institutions.

XIV. DISCUSSION AND CONCLUSIONS

The project, focusing on real-time data streaming and analysis of stock data using AWS services such as Kinesis, Lambda, and Elasticsearch, has successfully developed a dynamic, scalable system capable of managing high volumes of financial data. This achievement is showcased through a real-time dashboard that visualizes and analyzes stock market trends and patterns. Key accomplishments include the system's adaptability to handle the rapid influx of stock data and its proficiency in providing immediate financial insights through interactive and informative dashboards created with Kibana. Future enhancements could aim at optimizing costs, ensuring long-term data storage, bolstering security measures, and incorporating advanced analytics and machine learning for more nuanced financial analysis. This implementation not only demonstrates the power and versatility of cloud-based solutions in financial data handling but also lays a groundwork for future advancements in the field of real-time stock market data analysis and visualization.

REFERENCES

[1] Y. K. Gupta and N. Sharma, "Propositional Aspect between Apache Spark and Hadoop Map-Reduce for Stock Market Data," 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), Thoothukudi, India, 2020, pp. 479-483, doi: 10.1109/ICISS49785.2020.9315977.

[2] C. Lee and I. Paik, "Stock market analysis from Twitter and news based on streaming big data infrastructure," 2017 IEEE 8th International Conference on Awareness Science and Technology (iCAST), Taichung, Taiwan, 2017, pp. 312-317,doi:10.1109/ICAwST.2017.8256469. https://doi.org/10.1145/3269206.3272011

[3] Z. Peng, "Stocks Analysis and Prediction Using Big Data Analytics," 2019 International Conference on Intelligent Transportation, Big Data Smart City (ICITBS), Changsha, China, 2019, pp. 309-312, doi: 10.1109/ICITBS.2019.00081.

[4] D. -H. Shih, H. -L. Hsu and P. -Y. Shih, "A Study of Early Warning System in Volume Burst Risk Assessment of Stock with Big Data Platform," 2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), Chengdu, China, 2019, pp. 244-248,doi:10.1109/ICCCBDA.2019.8725738.

[5] F. Gürcan and M. Berigel, "Real-Time Processing of Big Data Streams: Lifecycle, Tools, Tasks, and Challenges," 2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Ankara, Turkey, 2018, pp. 1-6,doi:10.1109/ISMSIT.2018.8567061.
[6] J. Zhang, Y. Zheng, J. Sun and D. Qi, "Flow Prediction in Spatio-Temporal Networks Based on Multitask Deep Learning," in IEEE Transactions on Knowledge and Data Engineering, vol. 32, no. 3, pp. 468-478, 1 March 2020,doi:10.1109/TKDE.2019.2891537.
[7] G. Pal, G. Li and K. Atkinson, "Near Real-Time Big Data Stream Processing Platform Using Cassandra," 2018 4th International Conference for Convergence in Technology (I2CT), Mangalore, India, 2018,pp.1-7,doi:10.1109/I2CT42659.2018.9058101.

XV. Appendix

A. Abstract

The abstract conveys what we want to do and how we intend to achieve our objectives.It also explains why we chose this topic in the first place.

B. Motivation

The motivation section explains why we chose this specific topic and how we plan to solve the problem.

C. Methodology

The methodology describes our project's infrastructure as well as the project's progress step by step.

D. Deliverables

Please review the relevant section for deliverables.

E. Relevance to the course

Please review the relevant section for Relevance to the course.

F. Novelty Uniqueness

Please review the relevant section for Novelty Uniqueness.

G. Visualization

We collaborated as a team to write the report, as recommended.The report is our original work, and we have cited the references we used in the reference section.The visualizations can also be found in the slides and report.

H. Significance to the real world

We collaborated as a team to write the report, as recommended.The report is our original work, and we have cited the references we used in the reference section.The visualizations can also be found in the slides and report.

I. Code Walkthrough

We intend to explain our project pipeline during the presentation so that viewers can better understand what we have done in our project.

J. Report

All of these rubrics were followed, and the report is attached in both Latex and PDF format.Grammarly is used to check the report's contents for format, language, plagiarism, and grammar.

K. Version Control Use of git/github

We intend to include a github link with version control. The work that we did as a team for this project has been added to the github link.

L. Discussion / Q and A

The team will use the questions and answers time wisely in order to answer all of the questions in the final presentation, and a slide will be reserved at the end for discussion.

M. Lessons learned

We gained profound insights and developed a comprehensive understanding of various cloud-based services and tools during the course of our project, which focused on harnessing real-time stock market data for analysis and visualization. Our project not only made a contribution to the field of financial data analysis, but it also provided us with valuable learning opportunities. It provided us with hands-on experience with big data technologies and their applications in real-time environments, which we can apply to future endeavors in data science and analytics.

N. Analytics Component

We use Opensource service to do visualization and analytics analyzation.

O. Innovation

We performed many data cleaning steps which gave us a better accuracy. Our visualisations and EDA gave us insights which helped us for modeling. Used more models to see which models outperformed well and also many evaluation metrics which helped us to choose the best model for the given dataset

P. Teamwork

The team member roles are stated in report.

Q. Technical difficulty

Please review the relevant section for Technical difficulty.

R. Grammarly

Grammarly is another tool we use to improve our writing. The screenshot is shown.

S. Practiced pair programming

We worked together and on projects.We used Google Collab to implement virtual pair programming and included it in the report.We also used GitHub for pair programming, where we added the required code and report and collaborated.

T. Impact

Please review the relevant section for Impact.

U. GitHub

The source code for this project can be found on GitHub: https://github.com/mariapoulose94/realtimedatastreamingandanalysis

V. Practiced agile / scrum (1-week sprints)

We used Trello to manage our project, which allowed us to schedule tasks for each team member and keep an agile board. https://trello.com/b/KVn1d2UD/bigdataprojectgroup-4

W. Slides

According to the rubrics, we created slides for all of the topics covered in our project.The slides are concise and cover the most important aspects of our project.

X. LaTex Format

We used the Overleaf template. According to the rubrics, we used the IEEE LaTeX template.We created the report in both.tex and pdf formats.

Y. Creative presentation techniques

We created slides with the necessary screenshots, points, and images. I followed rubrics and included designer slides for each topic.

Z. Literature Survey

We have added a literature review of significant papers on real-time data streaming and analysis. Each paper was explained in terms of title, goal, algorithms used, and results. We have included this in the slides and report according to the rubrics.