Department of Computer Science and Engineering (Data Science)

Subject: Big Data Engineering (DJ19DSL604)

AY: 2022-23

Experiment 10

(Mini Project)

Aim: Design the infrastructure of a Big Data Application.

Tasks to be completed by the students:

Task 1: Choose a problem definition which requires handling Big Data.
Task 2: Design the data pipeline for your application.
Task 3: Deploy your project on a suitable platform.
Task 4: Test your application with different volume, variety and velocity of data.

Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Department of Computer Science and Engineering (Data Science)

Report on Mini Project

Big Data Engineering (DJ19DSL604) AY: 2022-23

STOCKS ANALYSIS AND PREDICTION

DEV PATEL

60009200016

MITALI CHOTALIA

60009200017

ALISTAIR SALDANHA

60009200024

Guided By

Prof. Pradnya Saval

CHAPTER 1: INTRODUCTION

Stock market is highly dynamic and volatile hence it becomes crucial to analyze real time data of stocks. Real time analysis allows investors and traders to make informed decisions based on up-to-date information. This project is intended to be a concise report to explain the big data technologies such as Spark and Docker. These technologies are used to perform real time data analysis of stocks. This project provides wholesome analysis of real time data of stocks which can help investors to track market trends as well as identify patterns. Additionally, it provides future predictions of the stocks movement which can help investors to minimize risk and maximize their daily returns. Reliance, HDFC, TCS, Bajaj Finance and Infosys are the five companies for which real time analysis of their stock data and further prediction is performed at varying intervals. There are several reasons why big data technologies like PySpark are useful for real time analysis of stock data. Scalability and Flexibility are two important aspects when it comes to real time processing of data which is taken care of by PySpark. That is, it can handle large amounts of data with ease, allowing for analysis of massive datasets in real-time and flexibility means it can be easily integrated with other big data technologies like Hadoop, hive and Kafka providing a flexible and customizable platform for real time stock analysis. PySpark also provides the ability to perform complex analyses on real-time data, such as machine learning models for predicting market trends or anomaly detection to identify irregularities in stock behavior. PySpark includes machine learning libraries like matplotlib, pandas, NumPy, etc. which can be used to build predictive models for stock analysis. Since PySpark caters to the demand of building prediction models, it is used for this project. Docker is used further which ensures that the application runs consistently across different environments and can be easily deployed on any infrastructure that supports Docker. Docker for deployments also makes it easier to manage dependencies and updates as containers can be easily updated and replaced without affecting the rest of the system.

CHAPTER 2: DATA DESCRIPTION AND ANALYSIS

The real time stocks data has been taken from yfinance which is a python library that allows easy access to stocks data of any company from Yahoo Finance. Yahoo Finance. It provides a simple way to download stock data from Yahoo Finance, and is commonly used in financial analysis. The library can retrieve data for multiple stocks at once, and offers a range of data points, including opening and closing prices, high and low prices, volume traded, and more. Additionally, yfinance also supports real-time data streaming for several stocks. The dataset of each stock includes five columns which are Date, Open, High, Low, Close, Adj Close and Volume. The data need not require any storage systems as processing is done on real time data using PySpark. Four essential factors are considered and calculated for the purpose of data analysis. They are:

1. **Average daily percentage change in price** - Average daily percentage change in price is a metric that measures the average daily fluctuation in the price of a stock over a period, expressed as a percentage. It is essential to analyze this metric as it is useful for investors and traders to understand the volatility of a stock which helps in making informed decisions about buying or selling it.
2. **Average Daily Price Range -** It is calculated as the difference between the high and low prices of a stock over a given period, typically a day, and then averaged over a specific time period. It provides insights into the level of price movement of a stock. Investors can use the Average Daily Price Range to assess the level of risk associated with each stock and adjust their portfolio accordingly.
3. **Largest daily price increase -** It is an essential metric which is calculated as the difference between Close and Open. A large price increase could indicate positive news or developments for the company, such as strong earnings or a major contract win. This information can help investors and traders to identify potential high-growth stocks and sectors, which may be attractive for longer-term investments.
4. **Largest daily price decrease -** It is an essential metric which is calculated as the difference between Open and Close. Analyzing the largest daily price decrease can help traders and investors in identifying potential risks and volatility in the market. It also helps them

understand current market trends and patterns. For example, if several stocks in a particular sector are experiencing large price decreases, it may indicate a broader trend or issue affecting that sector.

Overall, these four factors can provide important insights to investors about the stock's performance, volatility, and potential risks and returns.

CHAPTER 3: DESIGN OF DATA PIPELINE

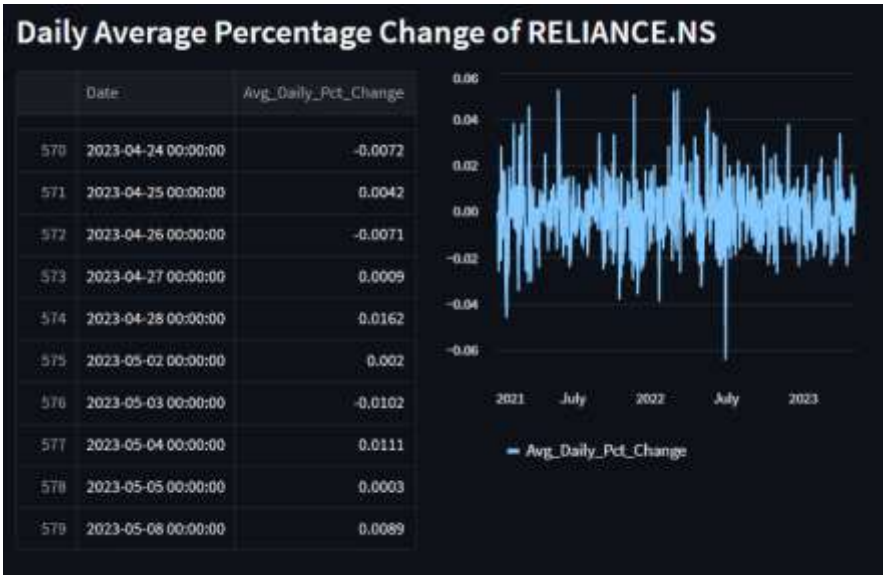
The data pipeline consists of the following steps:

- **Real-Time Stock Data**: This is the first step in the pipeline. Here, real-time stock data is obtained using the yfinance library based on the user's input for the stock symbol and interval.
- **Data Preprocessing**: The real-time stock data is then preprocessed and converted to a Spark DataFrame for further analysis. In this step, we extract the necessary columns from the DataFrame and clean the data by removing any null values.
- **Stock Price Model**: This step involves the training of the CNN (Conv1D) model to predict the stock prices. Here, we use Keras to build the model and train it using historical stock prices.
- **Data Analysis**: In the final step, we perform data analysis on the stock prices using Spark SQL. We compute the daily average price range and daily average percentage change in price for the selected stock symbol, and visualize the results using matplotlib and seaborn.
- **Deployment**: The code is deployed as a Docker container, which allows for easy deployment and management of the application. Also, the application is visualized using Streamlit. Streamlit is a powerful tool for creating interactive data visualizations and dashboards, and it can greatly enhance the user experience of data analysis.

Thus, the data pipeline is designed to provide real-time stock data, train a stock price prediction model, and perform data analysis on the historical stock prices.

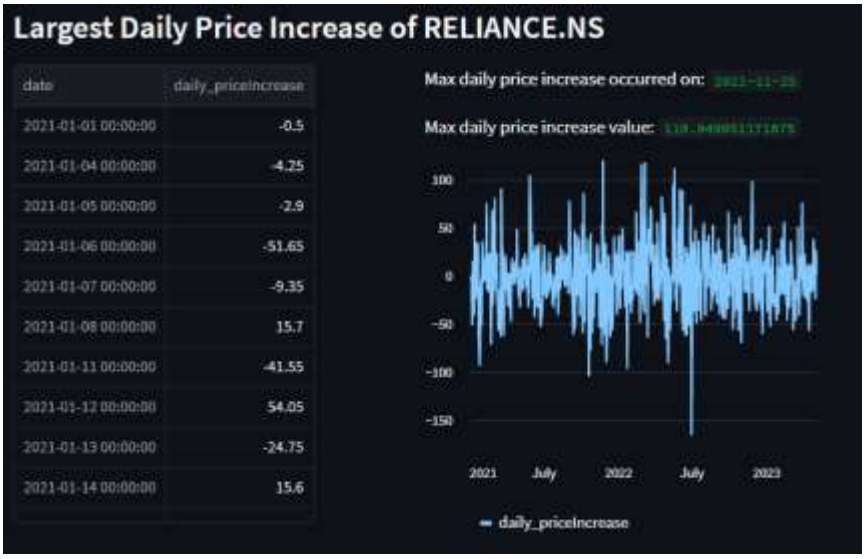
CHAPTER 4: RESULT ANALYSIS

The average daily price range for Reliance stock for the past 2 years gave an insight into the stock's volatility, potential risks and rewards associated with investing in the stock. Its consistent average daily price range indicates a lower degree of volatility, meaning that the stock's price will not fluctuate widely on a daily basis.

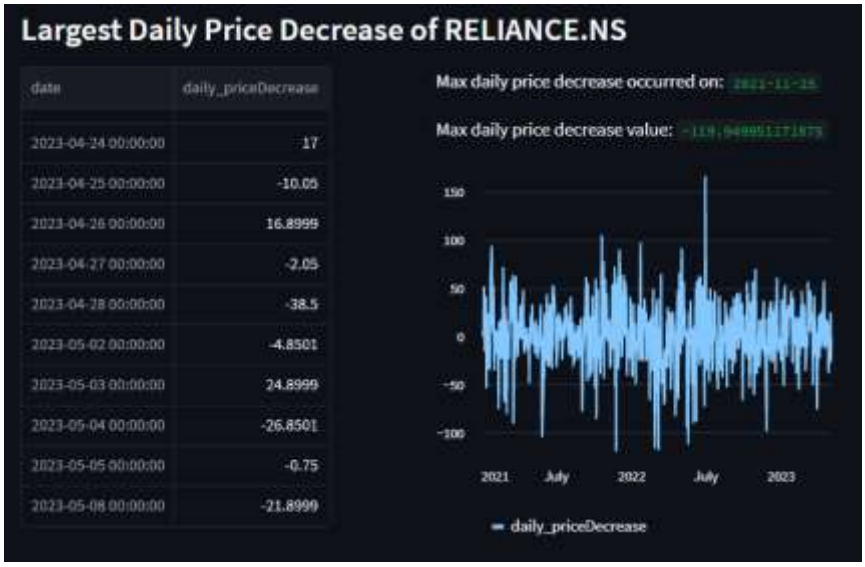


Based on the data provided by yahoo finance, the company appears to be quite volatile, with a mix of positive and negative daily percentage changes. Some days show significant drops in percentage, while others show a positive increase. We can see the highest percentage decrease is

in the month of July.



Highest daily price increase was calculated as Close - Open, where a large price increase could indicate a positive trend for the company, The highest daily price increase occurred on 25-11-2021, where the value was Rs 119.94. This type of information can be beneficial for traders and investors in identifying stocks and sectors that have the potential for significant growth. By analysing historical data on price movements, investors can make informed decisions on which stocks to invest in for the long term, thus maximizing their returns. This information can serve as a valuable tool for investors in identifying high-growth opportunities in the market.



Largest daily price decrease is calculated as Open - Close. Analysing the largest daily price decrease can help traders and investors in identifying potential risks and volatility in the market. It also helps them understand current market trends and patterns. For example, if we consider the daily price decrease for the Reliance stock, we see that it is in a negative trend till 8-5-2023 and the highest daily price decrease occurred on 25-11-2021 having a value of Rs -120 approximately.

Prediction for the Reliance Stock

We can see the plot for the prediction of 20 data points for the stock of Reliance using the CNN model based on the past data of the stock used for training the model.

CHAPTER 5: CONCLUSION AND FUTURE SCOPE

Thus, this Streamlit app performs analysis and prediction for a few Indian stocks. The app displays real-time stock data, predicts future prices using a machine learning model, and displays various analyses, such as daily average price range and daily average percentage change in price. The app uses the yfinance library to download stock data and Spark to perform the analysis. It also uses the Keras library to build a Convolutional Neural Network (CNN) to predict future stock prices. The app provides several options to the user to select the stock ticker and the interval of the data (1 minute, 1 day, 1 week, or 1 month). The user can also choose to show real-time data, stock prediction, or analysis of the stock. Overall, the app is a good demonstration of using machine

learning and Spark to perform stock analysis and prediction.

As for future scope, there are several possible directions to take this project:

- Improve the model performance: While the current model performs well, there is always room for improvement. One possible approach is to fine-tune the pre-trained model on the dataset or try different architectures to see if they can achieve better results.
- Expand the dataset: The current dataset consists of only five companies, and increasing the size and variety of the dataset could help the model to learn more robust features and improve its generalization performance.
- Develop more advanced web application: The current Streamlit application provides a simple interface for users to see the stock performance of the companies. However, there is scope to develop more advanced features such as interactive charts and graphs, additional input options for users, and a more user-friendly layout.
- Deployment to cloud services: While the above code demonstrates how to use Docker to deploy the application on a local machine, it can be deployed on cloud platforms such as AWS, Azure, or Google Cloud Platform to scale up the application and handle more traffic. Additionally, the deployment can be automated using continuous integration and continuous deployment (CI/CD) tools.