A project report on

TIME SERIES ANALYSIS DASHBOARD USING SPARK, GRAFANA AND STREAMLIT

Submitted by

Janani M

(Intern, Tech Mahindra)

DECLARATION

I hereby declare that the project entitled "TIME SERIES ANALYSIS DASHBOARD USING SPARK AND GRAFANA" submitted by me, for the award of the certification of my internship program provided by Tech Mahindra, Chennai, is a record of bonafide work carried out by me under the supervision of Mr. Manikandan B.

Place: Chennai

Date: 09/11/2023

PROOF OF WORK

Colab link:

https://colab.research.google.com/drive/1LU6jRJqhZGXGybdk0el0mmOCKMaWE1Ge?usp=sharing

Github link comprising of all the related documents:

https://github.com/itzjanani/TechM-Intership.git

ABSTRACT

This project proposes a new method for creating a dynamic Time Series Dashboard for Real-time Stock Price Prediction utilizing a stack that consists of Grafana for real-time data visualization, Apache Spark for machine learning and data processing, and Streamlit for creating interactive user interfaces. The system ingests, processes, and analyzes streaming stock price data by utilizing Apache Spark's distributed processing capability. This enables the creation and use of predictive models that enable real-time stock price forecasts. With Grafana, you can create dynamic dashboards that are configurable and show trends and projections for stock prices. But elaborate dashboard development and visualization in Grafana proved to be rather difficult, thus Streamlit was investigated as a more user-friendly option for web-based dashboard design. With Streamlit's user-friendly interface, traders and financial analysts may interact with models in an easy way and make well-informed decisions quickly. The benefits of these technologies are combined in this integrated solution to offer a responsive and all-encompassing system for real-time stock price prediction and decision support in the quick-moving financial markets.

CONTENTS

Introduction

The project "Time Series Analysis Dashboards with Spark, Grafana, and Streamlit" provides a flexible method for analyzing and visualizing time series data. The project integrates Apache Spark's capabilities for effective data processing and gets the dataset ready for analysis, all while keeping an eye on user-friendliness. Numerous important indicators are computed, such as the rolling mean, RSI, Bollinger Bands, and more. The project presents two approaches to visualization: Grafana, an open-source platform for power users, and Streamlit, which offers a simple and intuitive interface for creating dashboards quickly. This project gives users the ability to fully utilize time series data for data-driven decision-making by making it readable and understandable.

Going deeper into the exploration, we make use of time series analysis methods such as ARIMA modeling, which have the ability to reveal complex patterns in the data. This enhances the project's forecasting and predictive capabilities, making it a useful tool for individuals who wish to base decisions on historical data. Users can customize Streamlit to their own requirements by having the option to switch between Grafana's extensive data visualization and Streamlit's simplicity. In order to enable users to effortlessly conduct time series analysis and convert data into useful insights, regardless of their experience level or background, the project attempts to build a bridge between simplicity and sophistication.

PySpark

PySpark is frequently employed in time series analysis due to its capacity to efficiently process and analyze extensive time-ordered datasets. Its scalability and parallel processing capabilities make it suitable for tasks like feature engineering, model training, and forecasting, which often demand high computational performance. PySpark's integration with machine learning libraries, real-time streaming analysis, and compatibility with Python libraries for time series analysis provide a comprehensive environment for time series data manipulation and predictive modeling. The ability to work within the broader big data ecosystem further enhances its utility in managing and analyzing time series data, making it a valuable tool for various time-dependent applications, from financial forecasting to real-time monitoring.

1.1 WHAT IS PYSPARK?

PySpark is the Python library for Apache Spark, a powerful open-source framework for big data processing and distributed computing. PySpark provides an interface to use Spark with Python, enabling data engineers, data scientists, and developers to leverage the capabilities of Spark for data processing, machine learning, and analytics using the familiar Python programming language.

PySpark is a popular choice for data engineers and data scientists working on big data projects, as it provides the flexibility and performance of Spark while allowing them to leverage their Python programming skills. It is widely used in various industries for tasks such as data preprocessing, ETL (Extract, Transform, Load), machine learning, and real-time data processing.

1.2 DATASET

The Netflix Stock Prediction dataset is a vital resource for financial analysis and predictive modeling. It typically contains historical data of Netflix's stock prices, along with financial and economic indicators, essential for understanding stock performance. Researchers and analysts use this dataset for time series analysis, uncovering trends and patterns in stock prices, while data scientists and machine learning practitioners build predictive models to forecast Netflix's stock prices. Investors and portfolio managers assess risk and develop investment strategies using historical price data, and traders leverage it to make data-informed decisions. As the streaming industry evolves, this dataset gains importance in market analysis and investment strategies, offering valuable insights for decision-making in the dynamic field of finance.

Loading dataset

```
[29] df = spark.read.csv("/content/NFLX.csv", header=True, inferSchema=True)
```

```
df.show()
```

```
+----------+--------+--------+--------+--------+---------+---------+
|      Date|    Open|    High|     Low|   Close|Adj Close|   Volume|
+----------+--------+--------+--------+--------+---------+---------+
|2002-05-23|1.156429|1.242857|1.145714|1.196429| 1.196429|104790000|
|2002-05-24|1.214286|   1.225|1.197143|    1.21|     1.21| 11104800|
|2002-05-28|1.213571|1.232143|1.157143|1.157143| 1.157143|  6609400|
|2002-05-29|1.164286|1.164286|1.085714|1.103571| 1.103571|  6757800|
|2002-05-30|1.107857|1.107857|1.071429|1.071429| 1.071429| 10154200|
|2002-05-31|1.078571|1.078571|1.071429|1.076429| 1.076429|  8464400|
|2002-06-03|    1.08|1.149286|1.076429|1.128571| 1.128571|  3151400|
|2002-06-04|1.135714|    1.14|1.110714|1.117857| 1.117857|  3105200|
|2002-06-05|1.110714|1.159286|1.107143|1.147143| 1.147143|  1531600|
|2002-06-06|    1.15|1.232143|1.148571|1.182143| 1.182143|  2305800|
|2002-06-07|1.177857|1.177857|1.103571|1.118571| 1.118571|  1369200|
|2002-06-10|   1.135|   1.175|1.134286|1.156429| 1.156429|   484400|
|2002-06-11|1.156429|1.188571|1.128571|1.153571| 1.153571|  1003800|
|2002-06-12|1.153571|1.182143|1.089286|1.092857| 1.092857|  1799000|
|2002-06-13|1.104286|1.122143|    1.08|1.082857| 1.082857|  2567600|
|2002-06-14|1.082143|1.085714|    0.98|0.986429| 0.986429|  4783800|
|2002-06-17|1.013571|1.059286|0.917857|0.922143| 0.922143|  4855200|
|2002-06-18|0.938571|0.939286|0.842143|0.910714| 0.910714| 10808000|
|2002-06-19|0.911429|0.996429|0.892857|0.951429| 0.951429|  5482400|
|2002-06-20|0.971429|    1.04|    0.95|0.957143| 0.957143|  6111000|
+----------+--------+--------+--------+--------+---------+---------+
only showing top 20 rows
```

1.3 WHAT ARE INDICATORS?

In time series analysis, indicators are essential instruments for deriving meaningful insights from data point sequences captured at progressively longer intervals. The objective of these mathematical or statistical metrics is to identify underlying patterns, trends, and features in the time series data. Analysts and researchers can gain a better understanding of the dynamics and behavior of the data over time by using indicators to measure important characteristics such central tendency, dispersion, autocorrelation, seasonality, and stationarity. This information is essential for forecasting, finding anomalies, making well-informed decisions, and doing in-depth analysis in a variety of fields, such as finance, economics, climate science, and more. As a result, indicators are essential for revealing the insights concealed in time series data, improving our capacity to properly analyze and apply this priceless knowledge.

Some of these indicators include:

1. Rolling Mean: Also a moving average, it provides a smoothed representation of data over a rolling window to highlight trends.

2. Rolling Standard Deviation: Measures the variability or dispersion within a rolling window, helping to identify periods of increased or decreased volatility.

3. Simple Moving Average (SMA): SMA calculates the average of closing prices over a specified period. It's used to identify trends and smooth out price fluctuations.

4. Exponential Moving Average (EMA): EMA gives more weight to recent prices, making it more responsive to recent price changes, which can help traders react to trends more quickly.

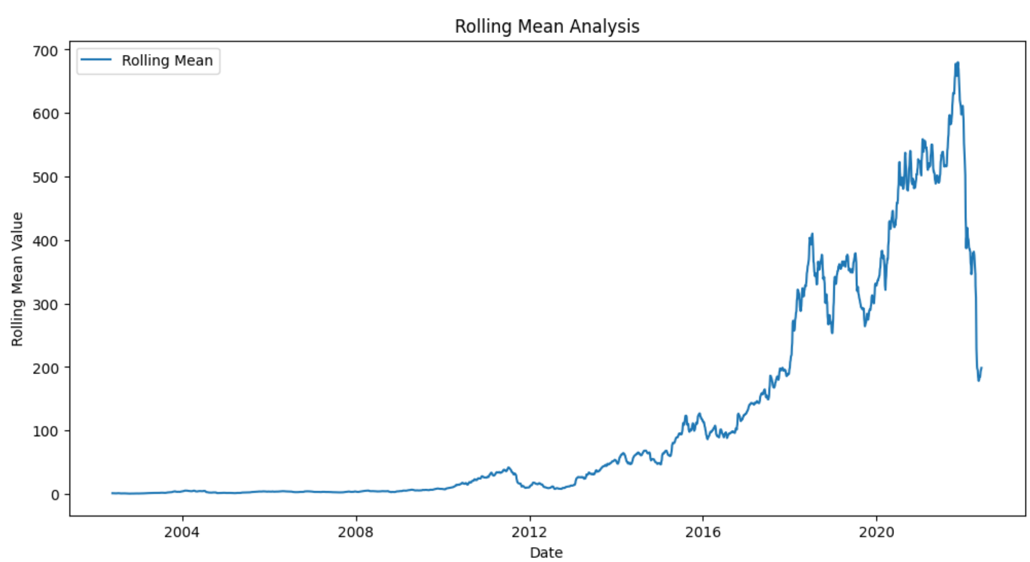
5. Bollinger Bands: Bollinger Bands consist of a middle band (SMA) and upper and lower bands that are typically two standard deviations away from the middle band. They help identify volatility and potential reversal points.

These technical indicators are widely used in technical analysis for making trading decisions and gaining insights into price trends, momentum, and potential reversals in financial markets. Traders often use a combination of these indicators to form a more comprehensive view of market conditions.

1.4 VISUALIZATION

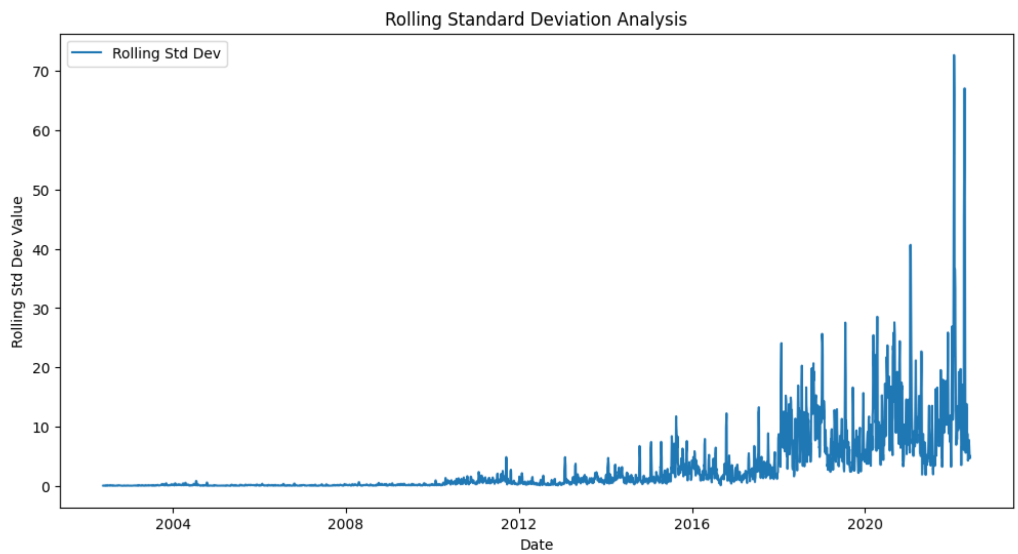
Rolling Mean Analysis:

The Rolling Mean is a straightforward moving average that is computed over a predetermined period of time and is used to mitigate brief price swings in the stock. We've made it simpler to see the overall trend in the Netflix stock price by plotting the Rolling Mean over time in this visualization. The Rolling Mean highlights upward or downward trends as it changes, capturing the general direction of the stock's movement.



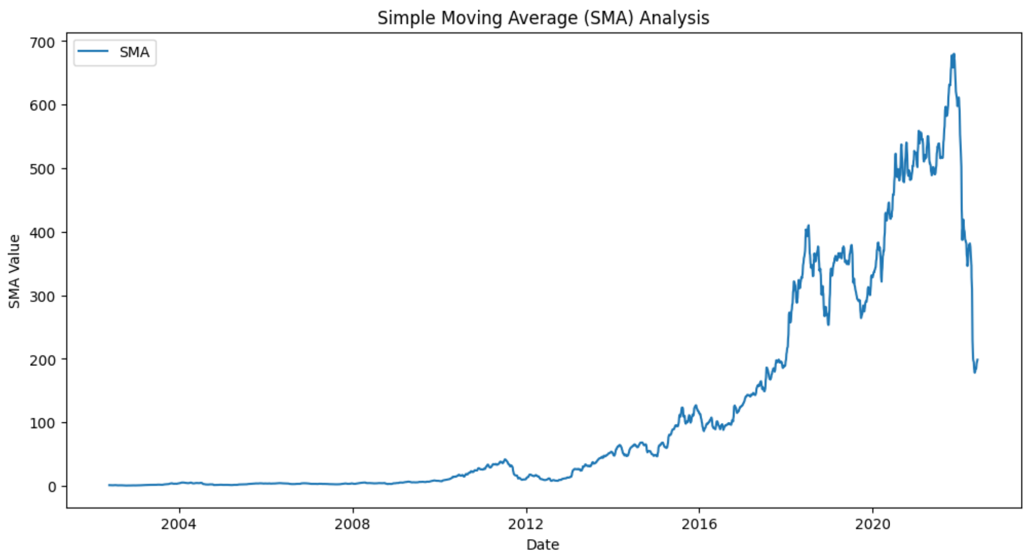
Rolling Standard Deviation Analysis :

This metric represents the volatility of prices. It aids in the understanding of a stock's level of risk by traders and investors. We have plotted the Rolling Standard Deviation against time in the visualization. Higher Standard Deviation values correspond to higher volatility, whereas lower Standard Deviation values correspond to lower volatility. This graph aids in evaluating Netflix stock's risk and possible profitability.



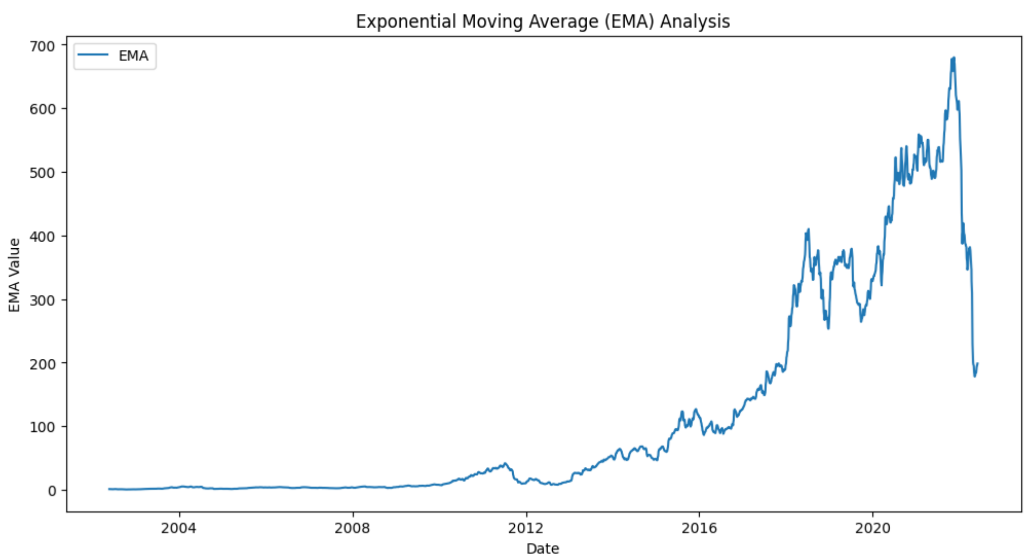
Simple Moving Average (SMA) Analysis:

Another trend-following indicator that helps spot trends and smooth out price data is the Simple Moving Average (SMA). We have produced a line chart that illustrates the SMA over time in this visualization. A bullish trend is indicated when the SMA line crosses above the stock price, and a bearish trend is indicated when it crosses below. The SMA is frequently used by traders to help them decide whether to buy or sell.



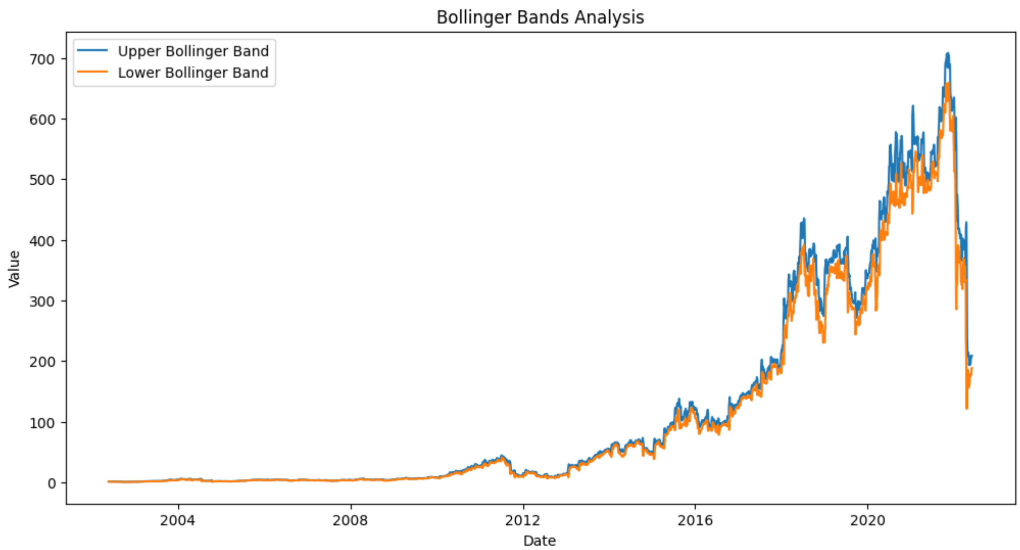
Exponential Moving Average (EMA) Analysis:

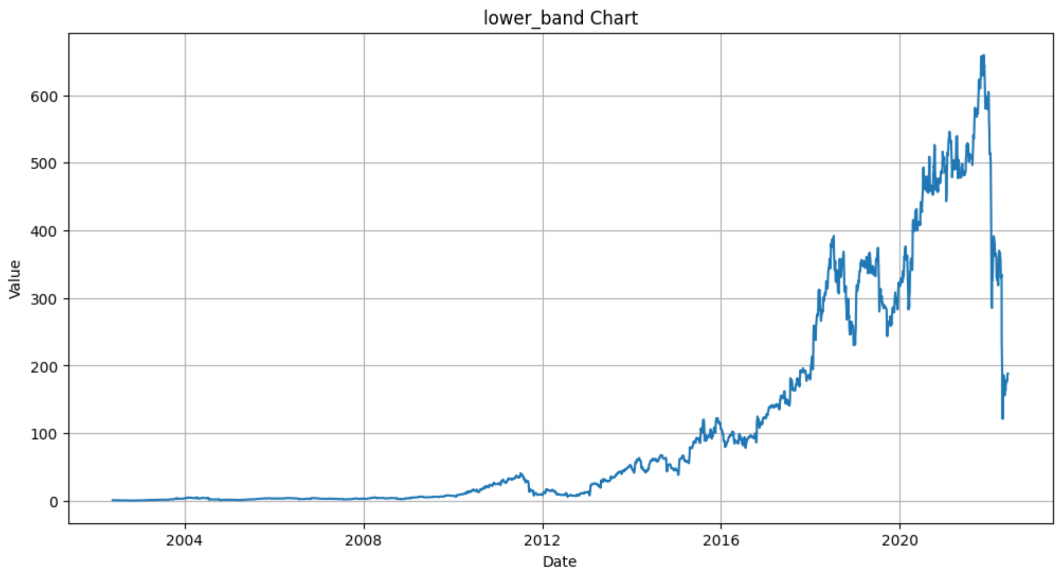
The EMA is comparable to the SMA but responds more quickly to recent price fluctuations because it places more weight on recent price data. The EMA is represented graphically by a line chart. An possible uptrend is suggested when the EMA crosses above the stock price, and a potential downtrend is indicated when it crosses below. The EMA is especially helpful in determining transient price changes.



Bollinger Bands Analysis:

Two standard deviation lines and a SMA make up a Bollinger Band. Both the Upper and Lower Bollinger Bands are plotted on the same chart in the visualization. The stock price may be overbought, indicating a possible reversal, when it gets close to the Upper Bollinger Band. On the other hand, if it gets close to the Lower Bollinger Band, it might be oversold, suggesting a potential reversal.





The trend, volatility, and possible buy/sell signals for the Netflix stock are all insightfully shown by these visualizations. These charts are a useful tool for traders and investors to use when making decisions based on indicator analysis and historical price data.

1.5 ARIMA MODEL

A statistical analysis technique called autoregressive integrated moving average, or ARIMA, makes use of time series data to forecast future trends or to get a deeper understanding of the data set. If a statistical model forecasts future values by using historical data, it is said to be autoregressive. An ARIMA model could, for instance, attempt to estimate a company's profitability based on historical periods or predict a stock's future pricing based on its historical performance.

The ARIMA model is a type of regression analysis that measures the strength of one dependent variable in relation to other variables that change. Instead of using actual values, the model looks at discrepancies between values in the series to forecast future movements in securities or the financial markets.

ARIMA model elements include:

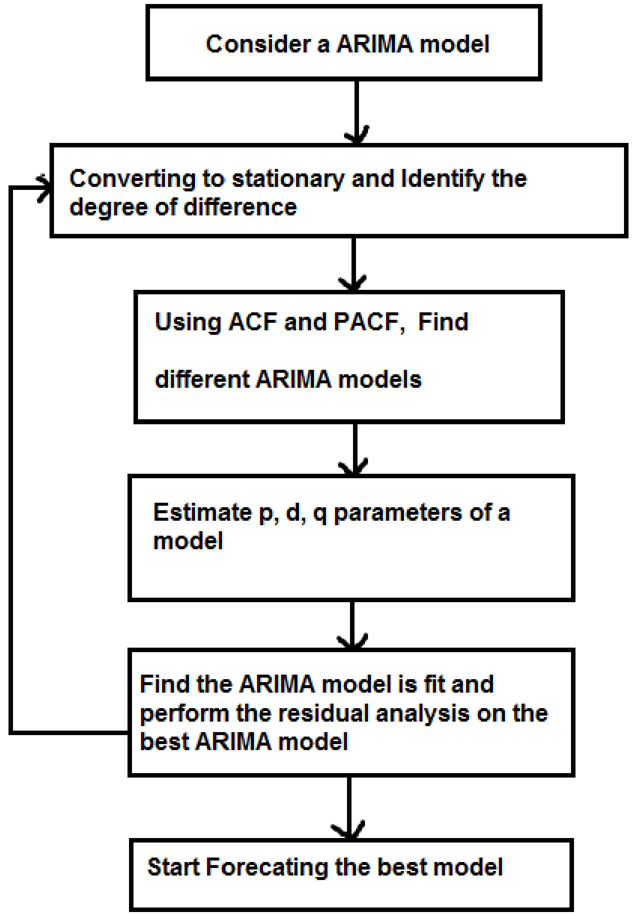
- Auto Regression (AR) model: A model that displays a variable that is evolving and regresses on its own lagged, or prior, values is known as an autoregression (AR) model.

- Integrated (I): The term "integrated" (I) refers to the process of differencing raw observations in order to enable the time series to reach a stationary state, wherein the data values are substituted with the difference between the current and prior values.
- Moving Average (MA): The dependency between an observation and a residual error from a moving average model applied to lagged observations is included into a moving average (MA).

ARIMA PARAMETERS

In ARIMA, every element serves as a parameter with a standard notation. In order to identify the type of ARIMA model being used, a conventional notation for ARIMA models would be ARIMA with p, d, and q, where integer values are used in place of the parameters. One definition of the parameters is:

- p: the lag order, or the number of lag observations in the model.
- d: the degree of differencing, or the number of times the raw observations are different.
- Q: the moving average window size, sometimes referred to as the moving average order.



Workflow of ARIMA model

APPLICATIONS FOR ARIMA

- Financial Time Series Analysis: To evaluate and forecast stock prices, currency exchange rates, and other financial data, financial markets frequently employ ARIMA models. These models are used by traders and investors to help them make wise judgments.
- Economic Forecasting: To forecast economic variables like GDP, inflation, and unemployment rates, economists use ARIMA models. Businesses and governments can make future plans with the aid of these forecasts.
- Weather Forecasting: Temperature, precipitation, wind speed, and other weather patterns are predicted by meteorologists using ARIMA models. For a number of industries, including transportation and agriculture, accurate weather forecasts are essential.

LIMITATIONS AND CHALLENGES

- Sensitivity to Parameters: The selection of p, d, and q affects the ARIMA models. Inaccurate forecasts may result from choosing the incorrect parameters.
- Assumption of Linearity: The underlying data in ARIMA models is assumed to follow a linear pattern. For nonlinear data, they might not function well.
- Not Appropriate for All Data: While real-world data frequently shows trends, seasonality, and other complex patterns that may call for more sophisticated models like SARIMA or Prophet, ARIMA models are designed for stationary data.

ARIMA models are an essential tool for time series analysis, to sum up. Their applications range from weather forecasting to finance by offering a structured framework for modeling and forecasting time-dependent data. Making the most of ARIMA models in practice requires understanding their components, selecting parameters carefully, and taking into account their limitations, even though they provide insightful and useful predictions. ARIMA is just one of many methods that analysts can use to extract meaningful information from time-dependent data in the rich and dynamic field of time series analysis.

1.6 PREDICTION

The result of time series analysis and modeling is forecasted data. When predicted data is plotted next to the historical series, it becomes evident how the future is anticipated to develop. Line charts that combine the predicted values and the original data are frequently used in time series forecasting. This makes it possible for decision-makers to evaluate the forecasts' dependability and accuracy. Plotting prediction intervals is another way to illustrate the degree of forecast uncertainty. Furthermore, it is easier for stakeholders to discern between past data and future projections when different colors or markers are used for actual and forecasted values. In addition to offering insights, these visualizations help with improved strategy development and decision-making.

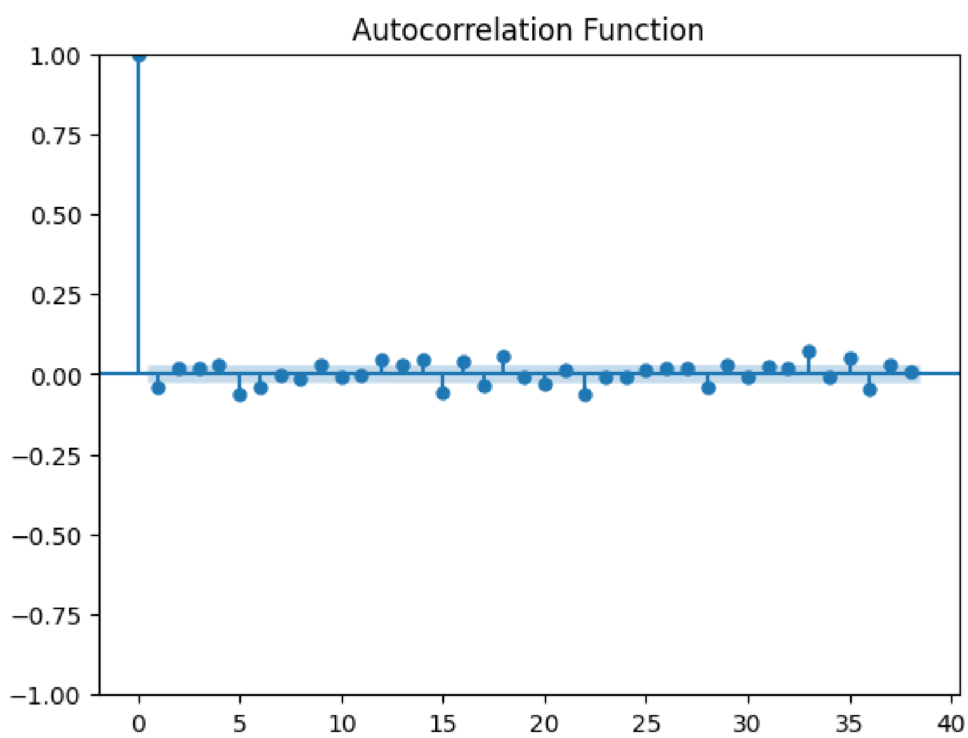
1. ADF Testing Outcomes:

ADF tests evaluate a time series' stationarity. An important indicator is the ADF Statistic, which indicates that the data may be stationary if it is significantly smaller than the critical values and the p-value is low (usually less than 0.05). Forecasts based on non-stationary data may not be accurate. Because it allows us to determine whether differencing is required to make the data stationary, the ADF test is essential.

```
ADF Statistic: -1.2470207311260355
p-value: 0.6530488146407217
Critical Values: {'1%': -3.4316556581508197, '5%': -2.862116959995152, '10%': -2.567077116457145}
ADF Statistic after differencing: -10.662460987641184
p-value after differencing: 4.371659316898826e-19
Critical Values after differencing: {'1%': -3.4316559188949403, '5%': -2.8621170751906178, '10%': -2.567077177780168}
```

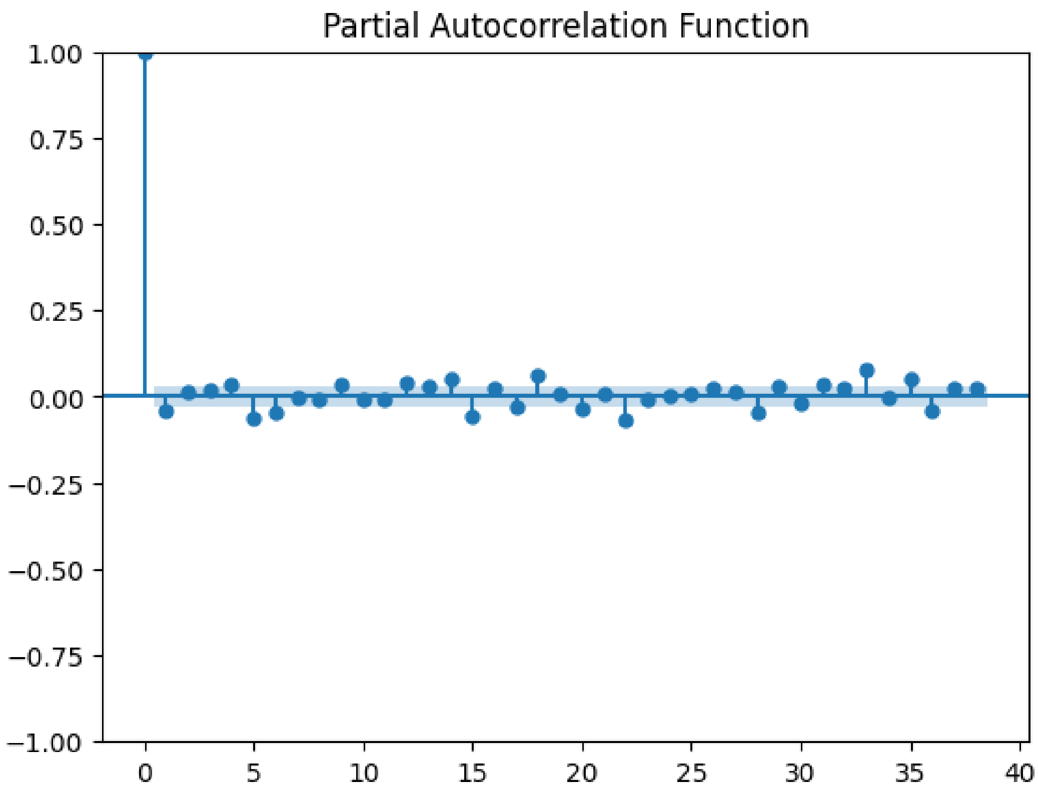
2. Autocorrelation Function (ACF) Plot:

A crucial component of time series analysis is the ACF plot. At various lag intervals, it shows the relationship between the series and its lags. The moving average component (q) in the ARIMA model can be ordered using the ACF graph for differenced data. The seasonality of the data is shown by the peaks and valleys in the ACF plot, which helps us determine the model order and the number of lags needed for the moving average.



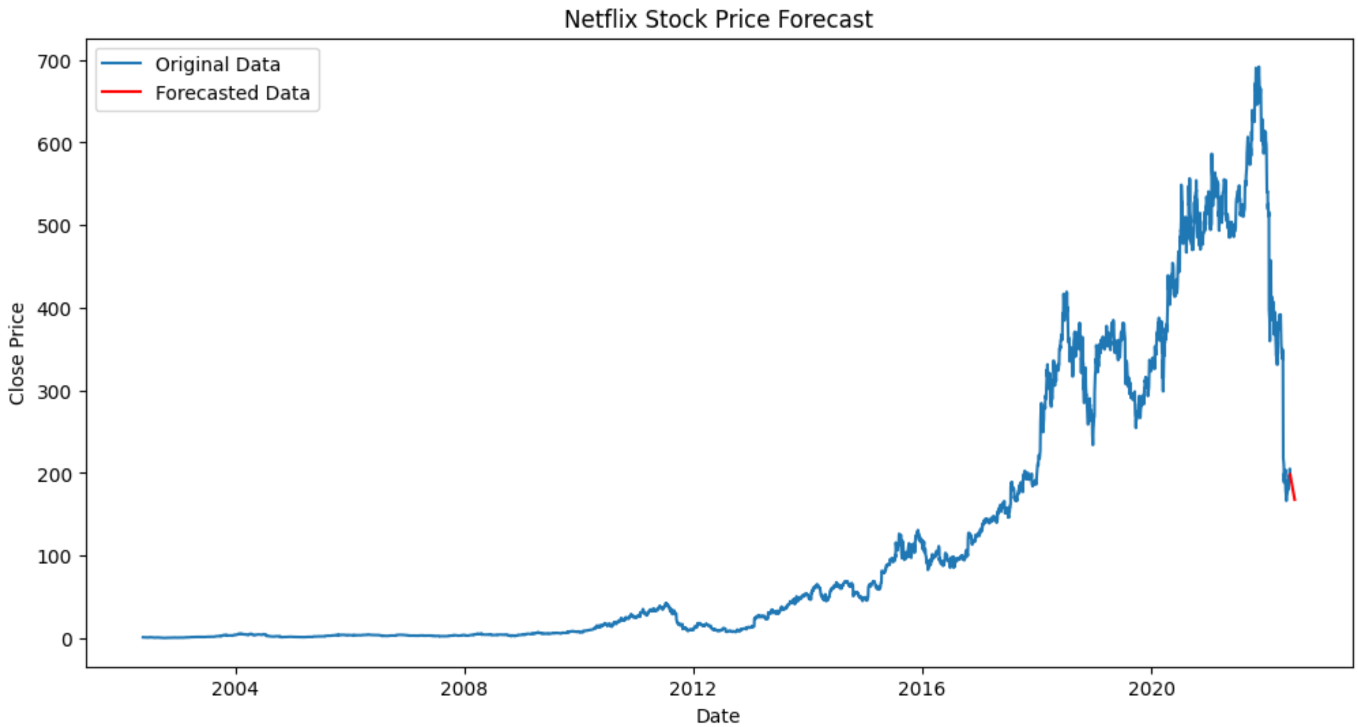
3. Partial Autocorrelation Function (PACF) plot:

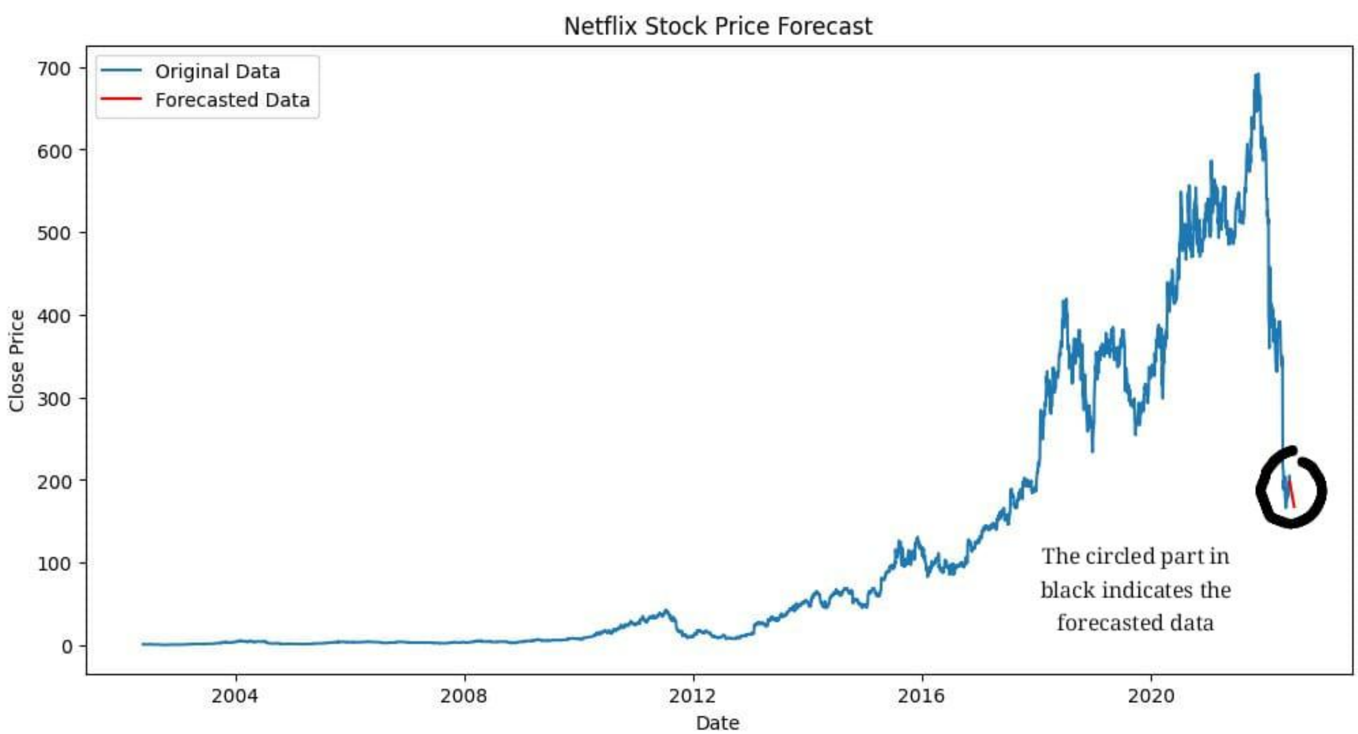
The ACF plot is enhanced by the Partial Autocorrelation Function (PACF) plot. By removing the impact of intermediate lags, it offers insights into the direct relationship between a data point and its lags. The order of the autoregressive component (p) in the ARIMA model is indicated by peaks in the PACF plot. To fully comprehend the proper model order, it is imperative to analyze both the PACF and ACF plots together.



4. ARIMA Forecast:

Using the ARIMA (Autoregressive Integrated Moving Average) model, the predicted values are displayed in the final graph. It's essential for explaining how the model forecasts upcoming data points. A visual depiction of the anticipated movement of the Netflix stock price is given by the red plotting of the forecasted values. Through the process of forecast-to-actual data comparison, stakeholders are able to assess the model's efficacy and make informed strategic decisions. The accuracy of these projections is essential for investment and financial planning.


Netflix Stock Price Forecast
Original Data
Forecasted Data
Close Price
Date
700
600
500
400
300
200
100
0
2004
2008
2012
2016
2020


Netflix Stock Price Forecast
Original Data
Forecasted Data
Close Price
Date
700
600
500
400
300
200
100
0
2004
2008
2012
2016
2020
The circled part in
black indicates the
forecasted data

STREAMLIT

A well-liked Python web application development library called Streamlit is useful for time series analysis. You can use Streamlit to create interactive applications and dashboards for the visualization and analysis of time series data. Among these are interactive charts for data visualization, widgets for user-driven data customization, preprocessing of the data, forecasting models for display, anomaly detection, performance metrics, and text annotations for insights. Because of its user-friendly interface and app sharing capabilities, Streamlit is a useful tool for streamlining and improving time series analysis procedures, enabling users to engage with and comprehend their data more successfully.

2.1 WHAT IS STREAMLIT?

Streamlit is an open-source Python library that makes it easy to create and share beautiful, custom web apps for machine learning and data science. In just a few minutes you can build and deploy powerful data apps.

A platform for easily interacting with time series data is provided by the user-friendly interface. Users have a more active role in the analysis process since they can easily change parameters and view data. Streamlit's direct data visualization feature makes it simple for users to spot patterns, track trends, and obtain insightful knowledge about the dataset. Time series analysis is becoming more accessible to a wider range of people, including analysts, data enthusiasts, and subject matter experts, thanks in large part to Streamlit's flexibility and agility. Its dynamic interface encourages a more intuitive interpretation of the information, facilitating well-informed decision-making across a range of industries, including healthcare and finance. The project gives users the ability to fully explore and unlock the potential of their time series data by utilizing Streamlit as the main visualization tool.

2.2 SOURCE OF DATA

The project sources its data from Yahoo Finance, a reputable and widely recognized financial data provider. Yahoo Finance offers a comprehensive repository of historical and real-time financial data, making it a preferred choice for data-driven projects. Its extensive coverage of financial instruments, including stocks, indices, and currencies, allows for a diverse range of analyses. Yahoo Finance's data is not only accurate but also reliable, meeting the standards of the finance industry.

Utilizing Yahoo Finance as the data source ensures the project's foundation is built on credible and trustworthy information. This is paramount for accurate time series analysis, as financial data integrity directly impacts the quality of insights and forecasts generated. Moreover, Yahoo Finance's user-friendly platform simplifies the process of data retrieval, enabling users to effortlessly access and integrate data into their analyses. Overall, the reliance on Yahoo Finance underscores the project's commitment to data quality and integrity, reinforcing its value as a reliable and informative time series analysis tool.

4.3 MODELS

The project implements a variety of machine learning algorithms to perform time series analysis and make forecasts, including:

1. Linear Regression: Linear regression is a fundamental algorithm that models the relationship between the dependent variable (in this case, time series data) and one or more independent variables (e.g., time steps or other relevant features). It's a simple yet powerful method for fitting a linear equation to the data and making predictions.

2. Random Forest Regressor: Random Forest is an ensemble learning method that combines multiple decision trees to create a robust and accurate predictive model. In time series analysis, it can capture complex relationships within the data and make reliable forecasts.

3. Extra Trees Regressor: Extra Trees is another ensemble method that builds multiple decision trees with random splits and features. It often performs well with noisy data and can be useful for handling time series data with irregular patterns.

4. K-Neighbors Regressor: The K-Neighbors Regressor is a non-parametric algorithm that makes predictions based on the values of nearby data points. It's particularly useful when dealing with local patterns and can be adapted to work well with time series data.

5. XGBoost Regressor: XGBoost is a popular gradient boosting algorithm known for its efficiency

and accuracy. It excels in capturing intricate patterns and is widely used in various data analysis tasks, including time series forecasting.

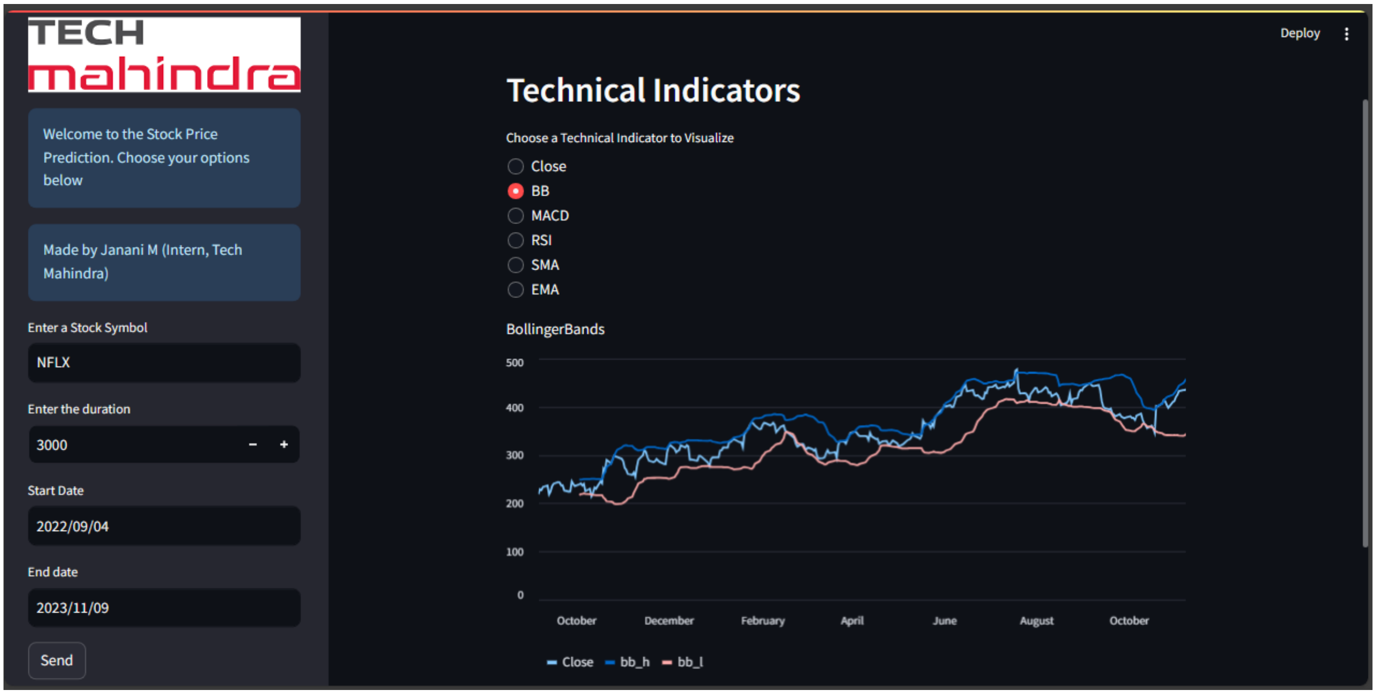
These machine learning algorithms are applied in the project to analyze historical time series data and make forecasts. Each algorithm has its strengths and can be tuned to suit the specific characteristics of the data, providing a comprehensive analysis and forecasting framework for users.

2.4 VISUALIZATION

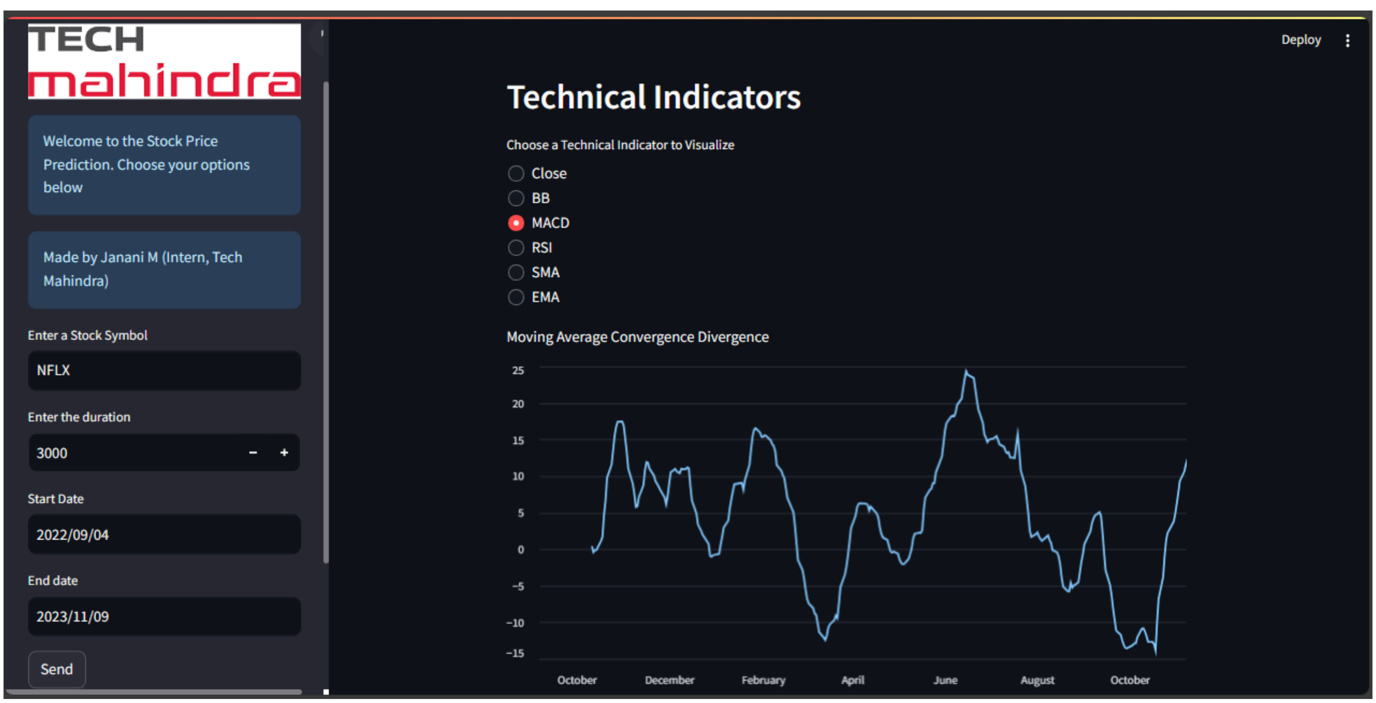
The project utilizes a set of technical indicators to visualize and analyze time series data. These indicators include:

1. Close Price: The closing price of a stock or asset is a fundamental indicator used in time series analysis. It represents the final price of the asset at the end of a given time period.

2. Bollinger Bands (BB): Bollinger Bands are volatility bands placed above and below a moving average. They help identify potential price reversals and overbought or oversold conditions.

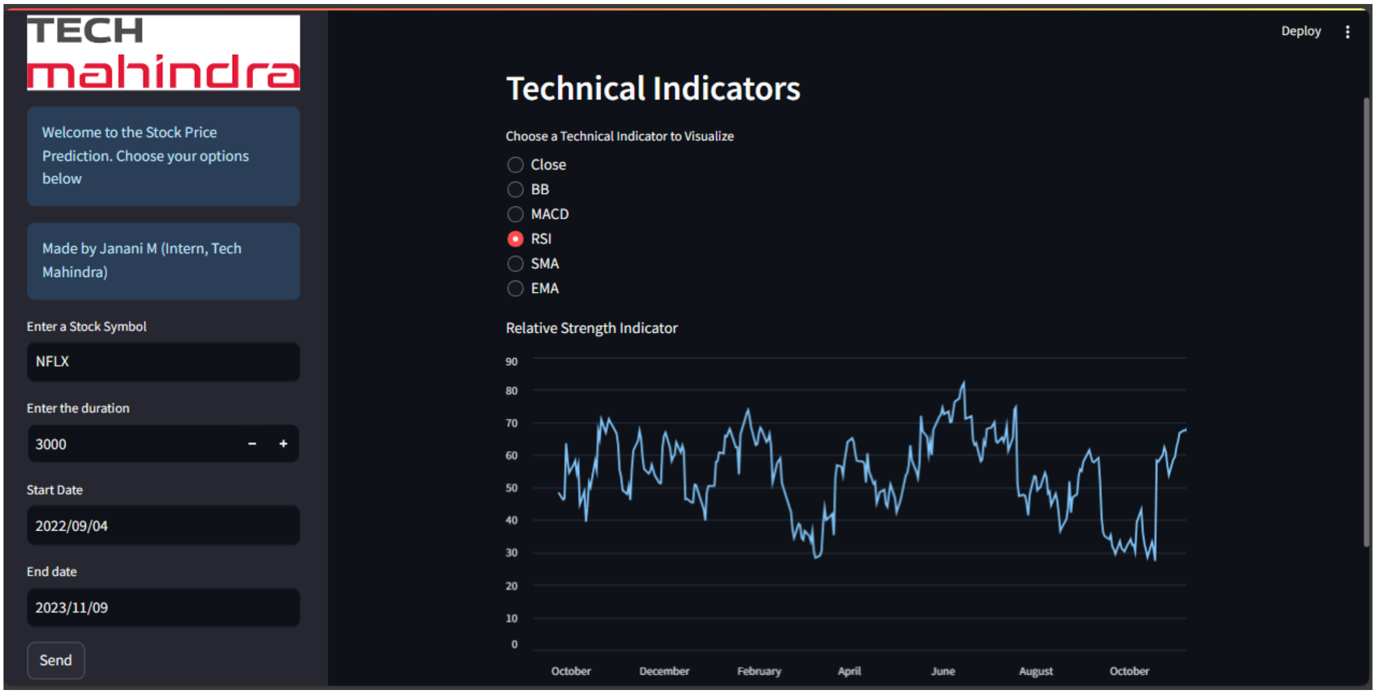


3. Moving Average Convergence Divergence (MACD): MACD is a trend-following momentum indicator that shows the relationship between two moving averages of an asset's price. It's used to identify changes in the strength, direction, momentum, and duration of a trend.

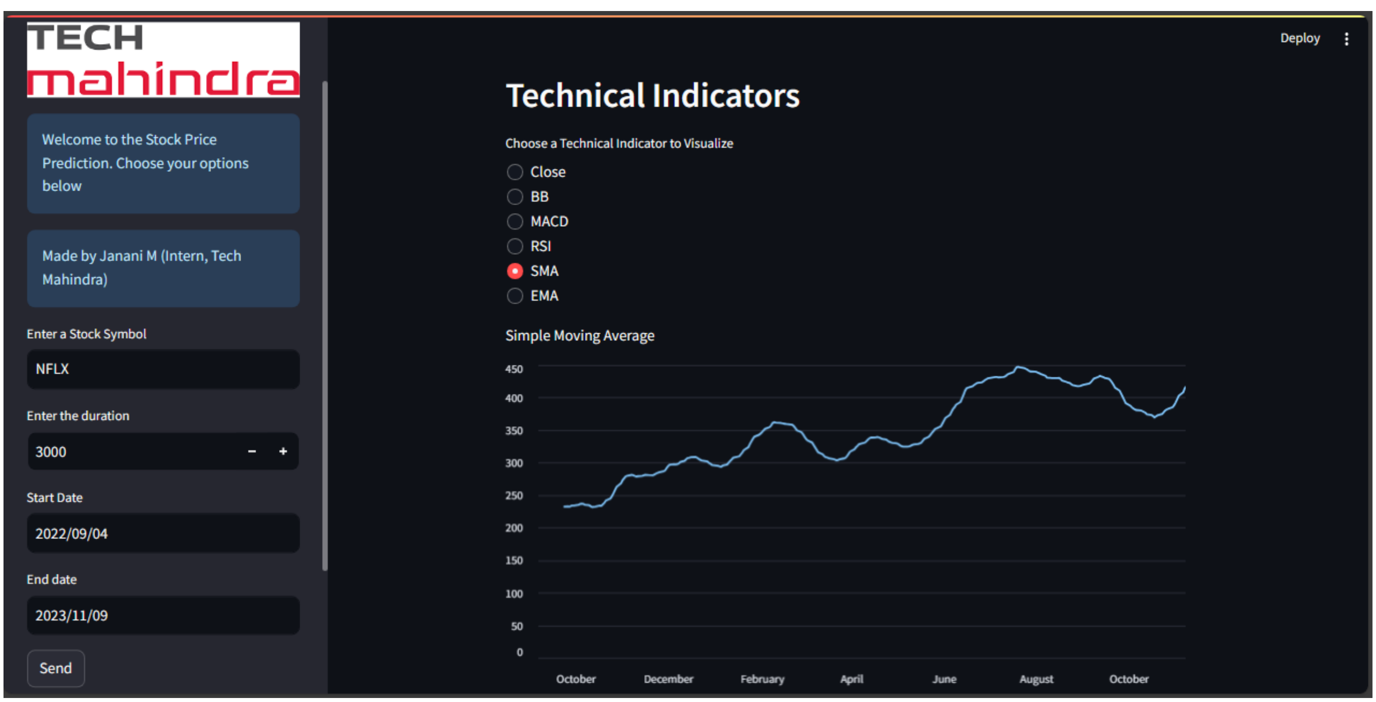


4. Relative Strength Index (RSI): RSI measures the speed and change of price movements and is

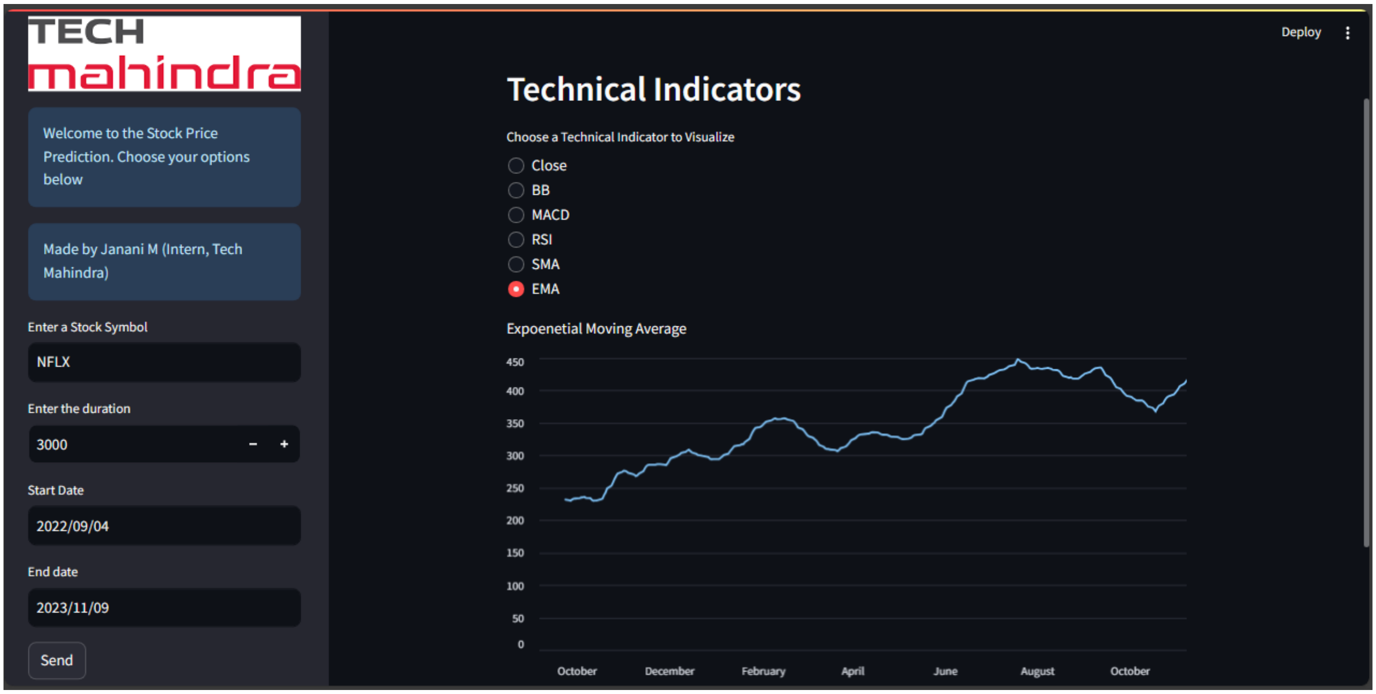
often used to identify overbought or oversold conditions in a market.



5. Simple Moving Average (SMA): The SMA is a straightforward indicator that calculates the average of a set of prices over a specific time period. It's used to smooth out price data and identify trends.



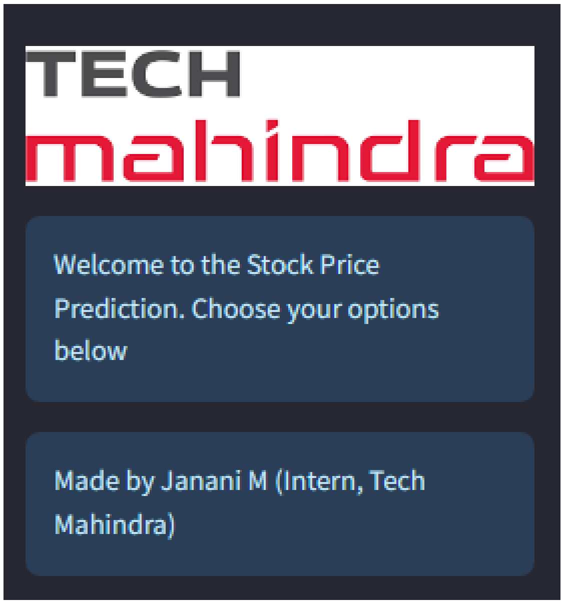
6. Exponential Moving Average (EMA): EMA is a weighted moving average that gives more weight to recent prices. It reacts more quickly to price changes and is useful for identifying short-term trends.

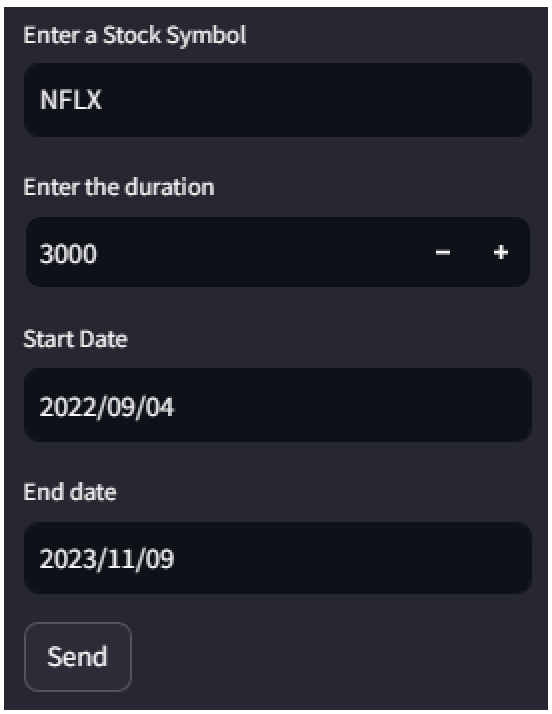


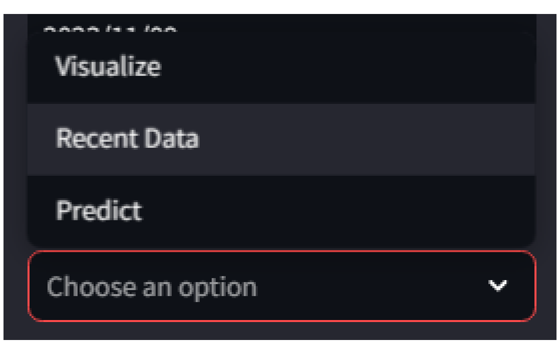
These technical indicators are applied to Netflix Stock Prediction data in the project to provide insights into market behavior, trends, and potential turning points. The combination of these indicators enables users to make informed decisions and forecasts regarding the asset's future price movements.

2.5 WEB PAGE

Navigating and exploring data is made easy by the responsive design and real-time updates. Users can easily navigate between indicators, engage with the data, and make data-driven decisions with Streamlit's dynamic charts and graphs. This integration makes the dashboard much more accessible and user-friendly overall by greatly enhancing the visualization capabilities.


TECH
mahindra
Welcome to the Stock Price Prediction. Choose your options below
Made by Janani M (Intern, Tech Mahindra)


Enter a Stock Symbol
NFLX
Enter the duration
3000
Start Date
2022/09/04
End date
2023/11/09
Send


Visualize
Recent Data
Predict
Choose an option

Recent data:

Make a choice

Recent Data

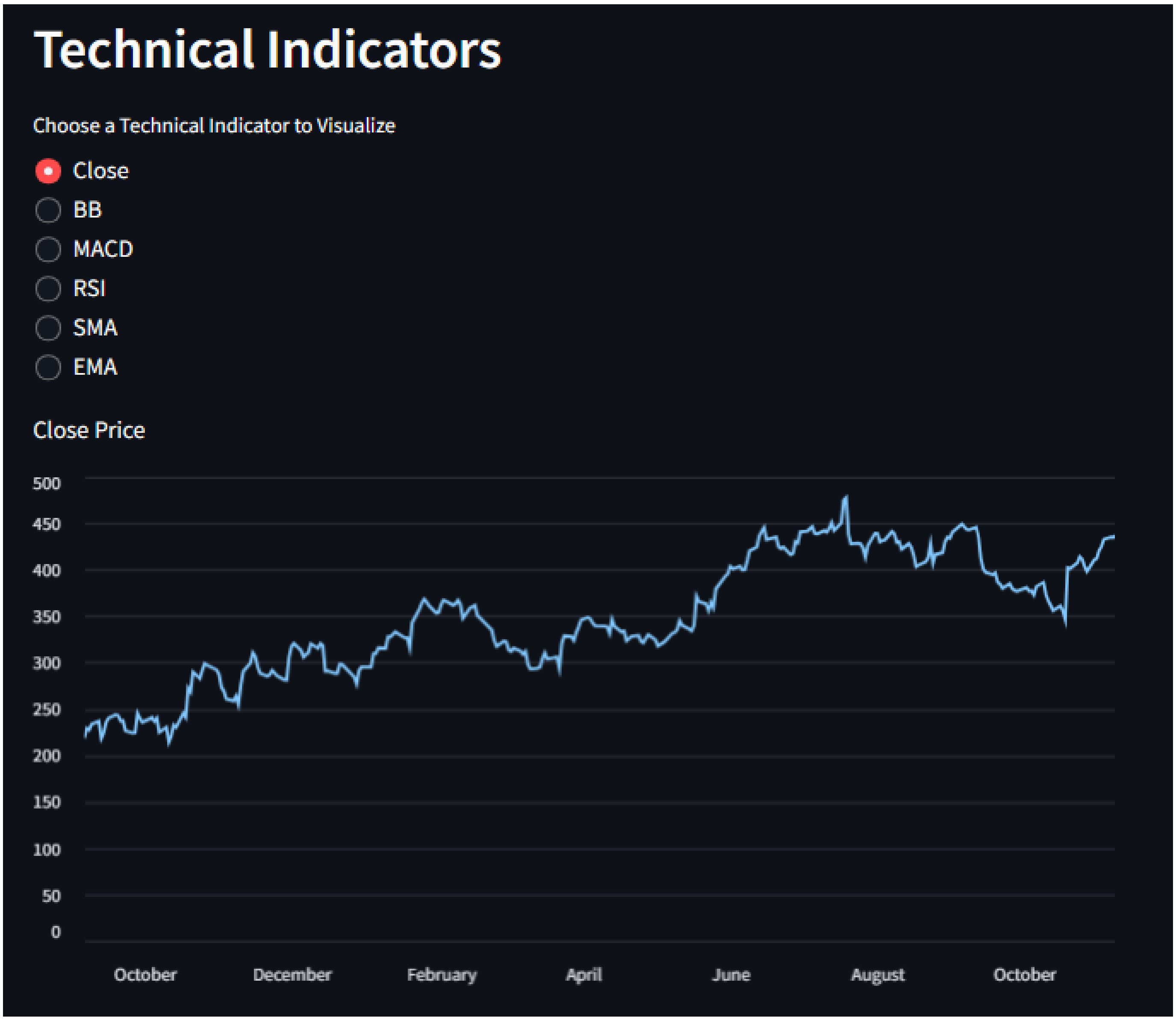
Stock Price Predictions

Recent Data

Date	Open	High	Low	Close	Adj Close	Volume	bb_h	bb_l
2023-10-26 00:00:00	411.42	417.31	401.54	403.54	403.54	6,849,700	419.6846	341.1734
2023-10-27 00:00:00	406.42	410.21	395.62	397.87	397.87	4,997,600	421.3941	341.4909
2023-10-30 00:00:00	402.35	412.82	399.41	410.08	410.08	5,317,100	424.7756	341.0844
2023-10-31 00:00:00	409.24	412.52	404.63	411.69	411.69	3,877,600	428.2273	341.1267
2023-11-01 00:00:00	414.77	420.6	414.18	420.19	420.19	4,806,100	432.8636	340.8194
2023-11-02 00:00:00	421.17	426.69	417.1	424.71	424.71	4,476,000	437.7905	341.1045
2023-11-03 00:00:00	428.76	434.82	425.53	432.36	432.36	3,664,800	443.6319	340.3481
2023-11-06 00:00:00	434.38	435.03	429.61	434.74	434.74	3,003,200	449.2136	339.6454
2023-11-07 00:00:00	436.18	437.64	431	434.61	434.61	3,291,100	454.0405	340.9475
2023-11-08 00:00:00	435	438.07	433.68	436.65	436.65	2,350,800	458.0815	343.9785

Visualize:


Make a choice
Visualize

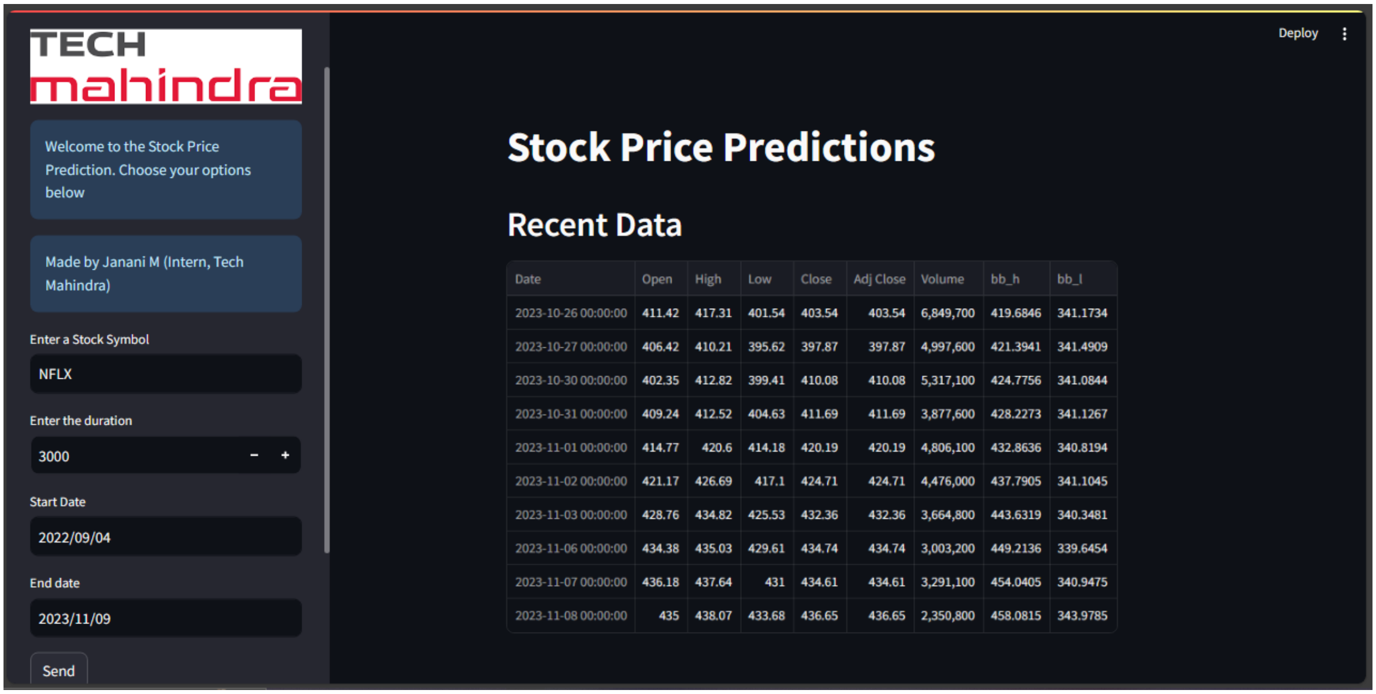

Technical Indicators
Choose a Technical Indicator to Visualize
Close
BB
MACD
RSI
SMA
EMA
Close Price
500
450
400
350
300
250
200
150
100
50
0
October
December
February
April
June
August
October

Predict:


Make a choice
Predict

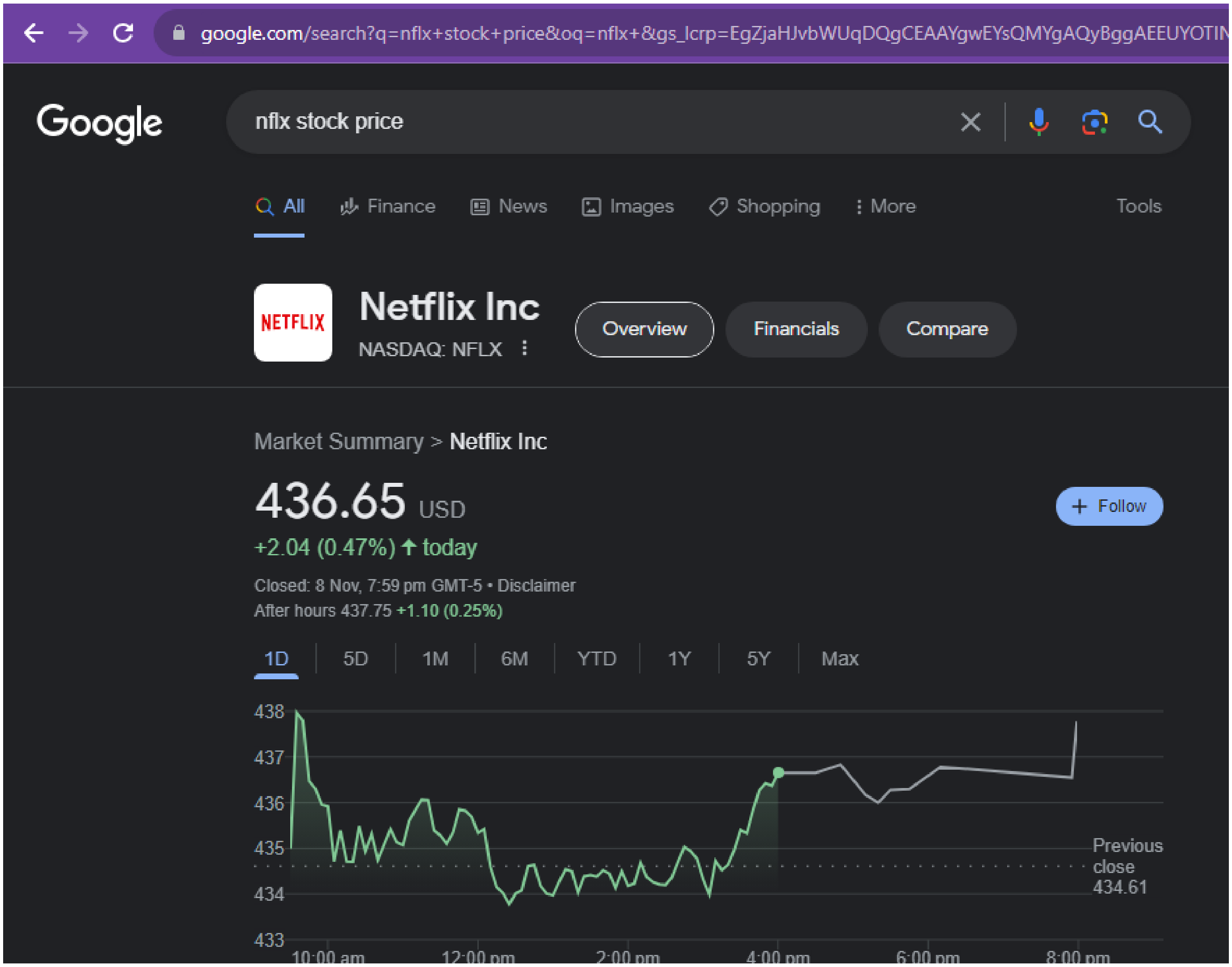

Stock Price Predictions
Choose a model
LinearRegression
RandomForestRegressor
ExtraTreesRegressor
KNeighborsRegressor
XGBoostRegressor
How many days forecast?
5
Predict

WEBSITE LANDING PAGE :



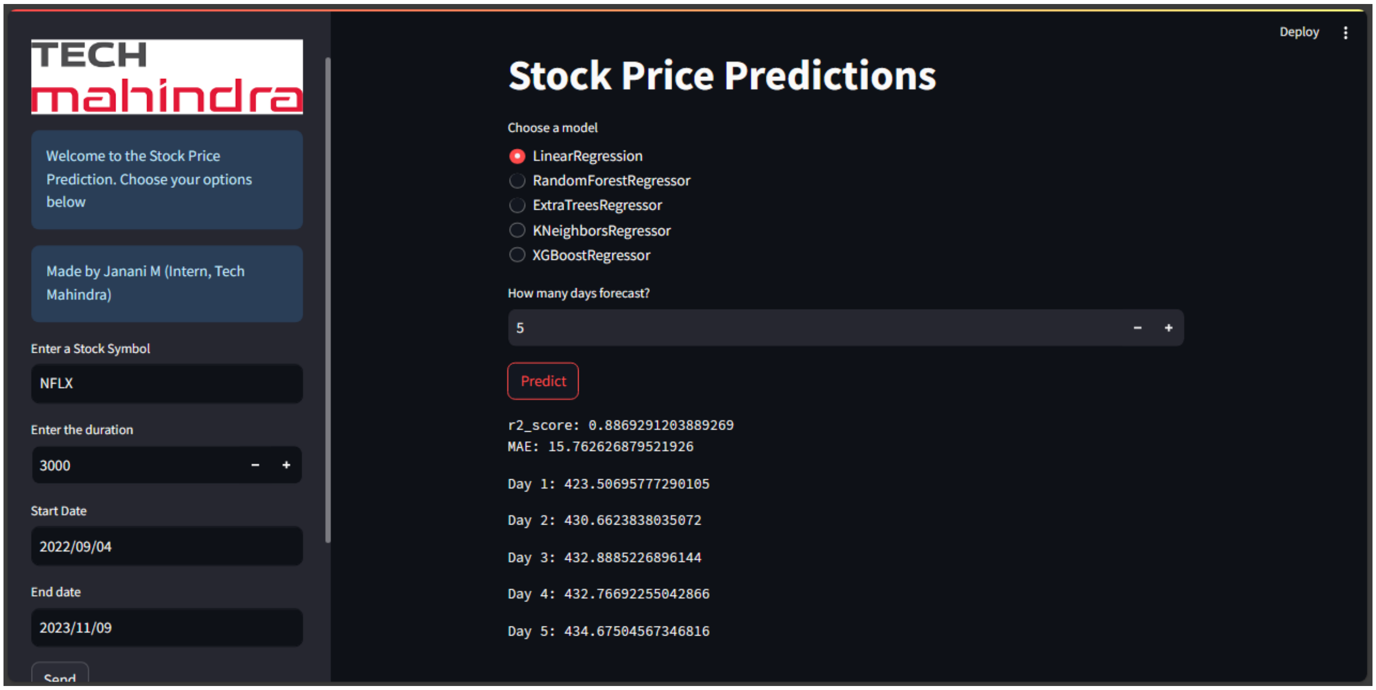
Result

Today's Netflix stock price is

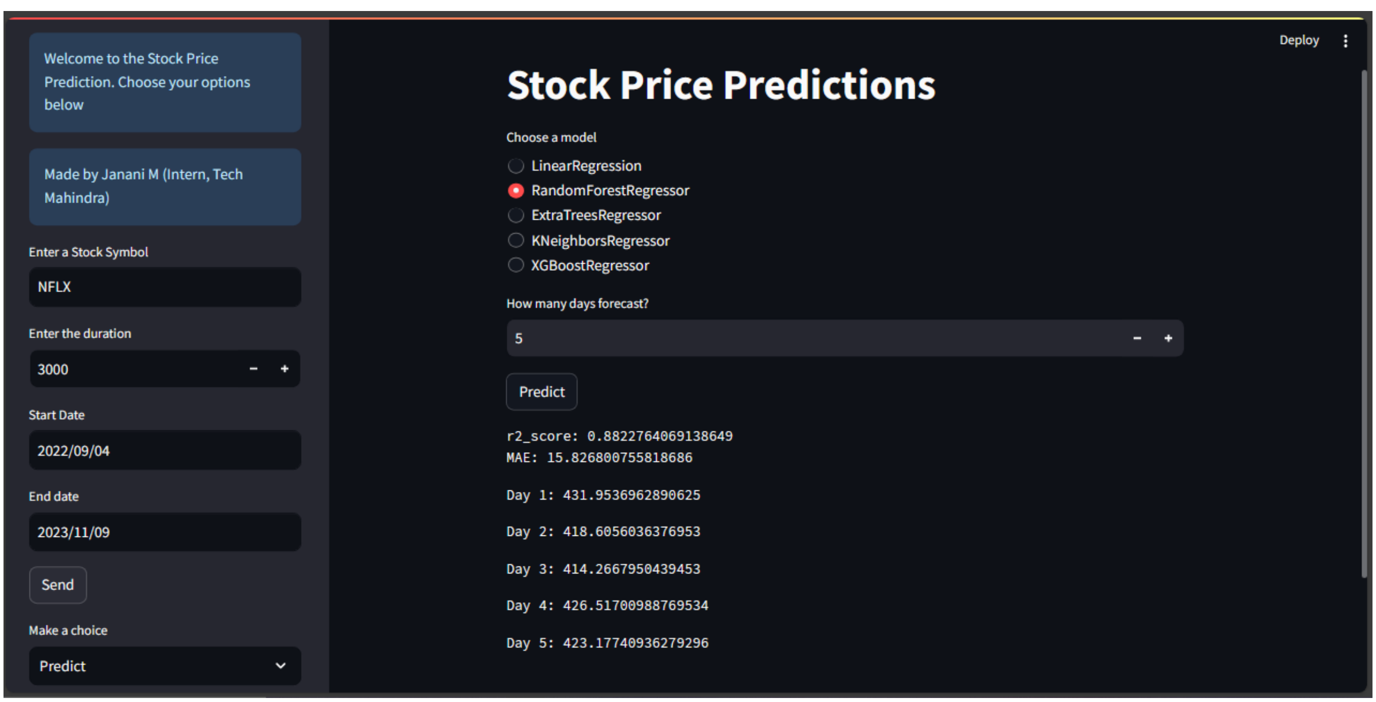


The project implements a variety of machine learning algorithms to perform time series analysis and make forecasts for the next five days using multiple ML models.

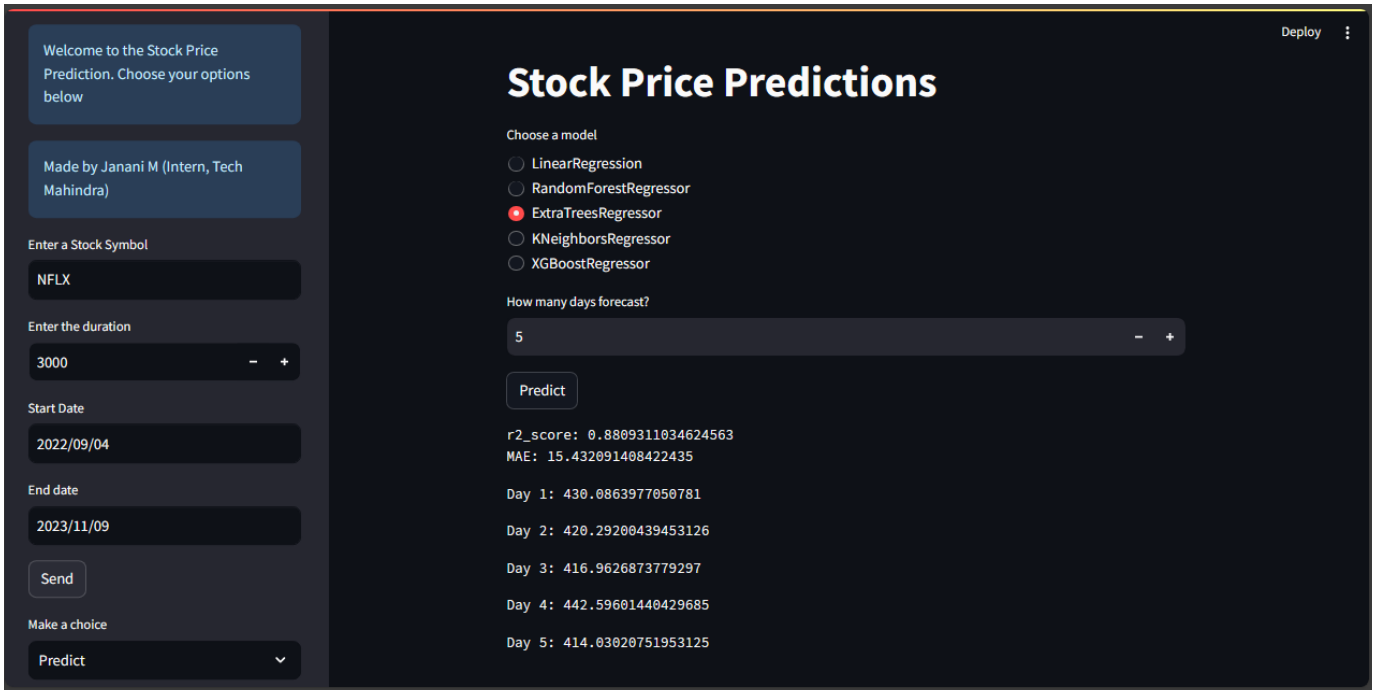
1. Linear Regression:



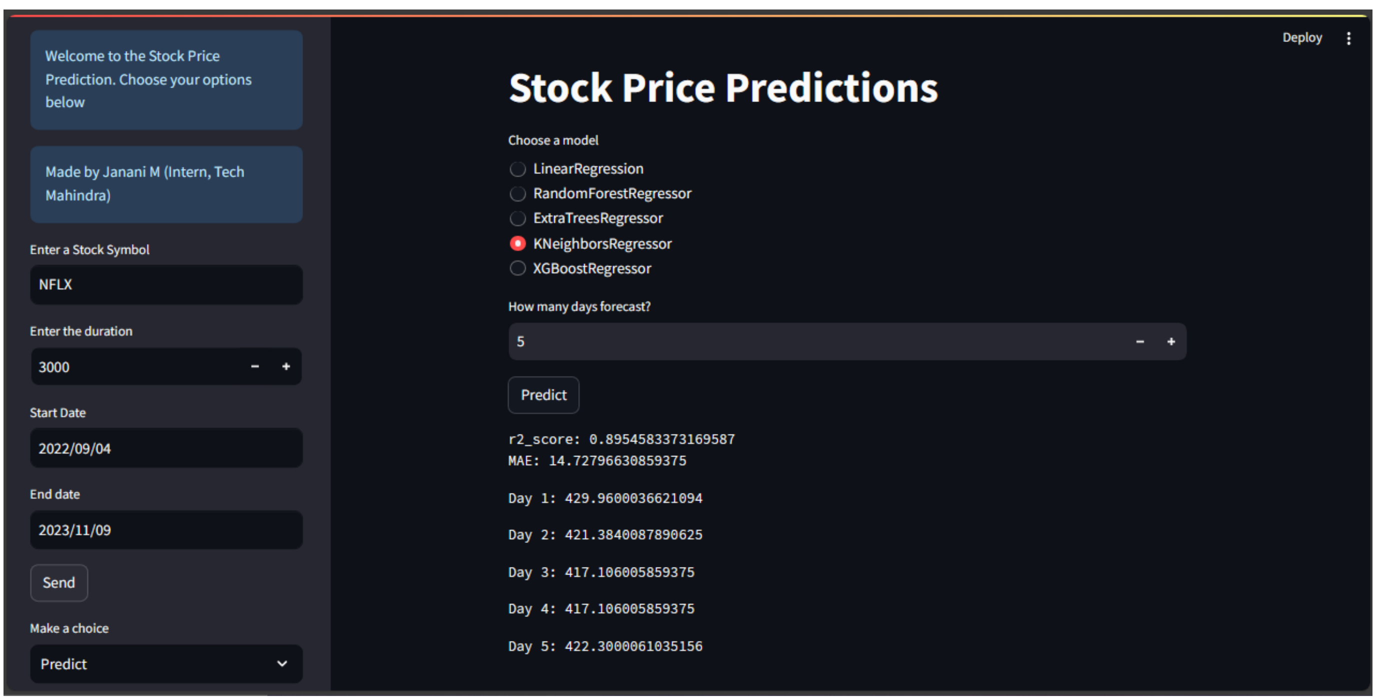
2. Random Forest Regressor:



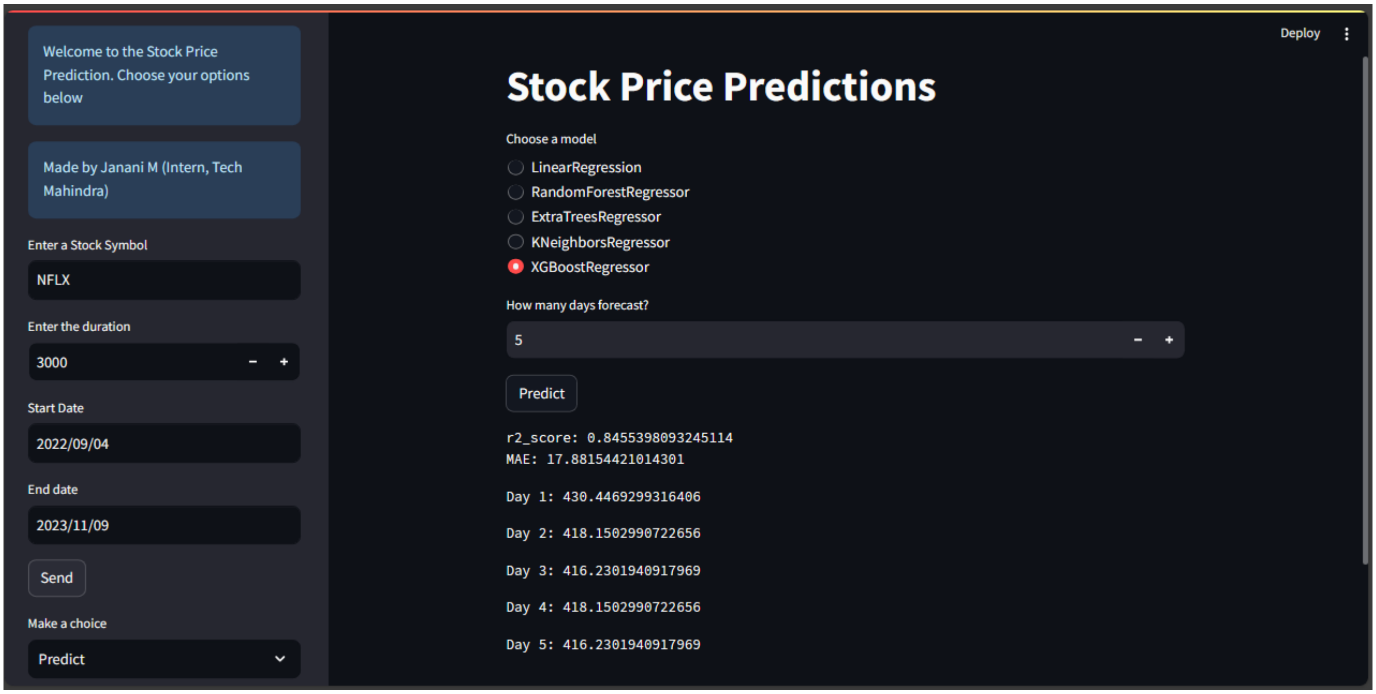
3. Extra Trees Regressor:



4. K-Neighbors Regressor:



5. XGBoost Regressor:



These machine learning algorithms are applied in the project to analyze historical time series data and make forecasts. Each algorithm has its strengths and can be tuned to suit the specific characteristics of the data, providing a comprehensive analysis and forecasting framework for users.

CONCLUSION

In conclusion, the "Time Series Analysis Dashboards with Spark and Streamlit" project provides a comprehensive and user-friendly platform for analyzing historical stock market data. By leveraging Spark for data processing and Streamlit for interactive web applications, the project ensures that users can effortlessly explore and understand complex time series data.

With the integration of technical indicators such as Close, Bollinger Bands (BB), Moving Average Convergence Divergence (MACD), Relative Strength Index (RSI), Simple Moving Average (SMA), and Exponential Moving Average (EMA), users can gain valuable insights into market trends and make data-driven decisions. The comparison between Grafana and Streamlit reveals that Streamlit's dynamic and interactive charts offer a superior user experience for data exploration.

Moreover, the project's source data from Yahoo Finance ensures the reliability and accuracy of the financial data used for analysis. The addition of machine learning models, such as Linear Regression, RandomForestRegressor, ExtraTreesRegressor, KNeighborsRegressor, and XGBoostRegressor, enables users to make more informed predictions and forecasts.

The inclusion of time series forecasting using ARIMA models further enhances the project's capabilities, allowing users to predict future stock prices with accuracy. These forecasts are presented through Streamlit's responsive web interface, making the analysis more accessible to a broader audience.

Overall, the "Time Series Analysis Dashboards with Spark and Streamlit" project offers a comprehensive and user-centric solution for financial data analysis, visualization, and forecasting, making it a valuable tool for investors and analysts in the stock market.