Prediction of Occurrence of Circuit Breakers in Stock Prices

Hetvi Desai
MSc (Data Science)
DAIICT
Gandhinagar, Gujarat
202218012@daiict.ac.in

Shreya Arora
MSc (Data Science)
DAIICT
Gandhinagar, Gujarat
202218032@daiict.ac.in

Dhruv Solanki
MSc (Data Science)
DAIICT
Gandhinagar, Gujarat
202218053@daiict.ac.in

Jatan Sahu
MSc (Data Science)
DAIICT
Gandhinagar, Gujarat
202218061@daiict.ac.in

***Abstract*—For countless years, researchers and scholars have been fascinated by how the stock market can be so volatile. When prices fluctuate uncontrollably, circuit breakers are used to control losses. These circuit breakers pause trading temporarily when a stock price crosses certain limits. In our study, we aim to create an algorithm that can predict when these circuit breakers might happen in the near future. In the initial phase, we have successfully devised a basic algorithm that utilizes historical data to identify instances of circuit breakers and generate the target feature. In the current phase, we aim to delve deeper into the subject by experimenting with diverse machine learning algorithms like Random Forest, Gradient Boosting, and Prophet, as well as employing a time-series model. This phase will heavily involve fine-tuning hyperparameters and meticulously evaluating the performance of the models. By combining these approaches, we hope to gain valuable insights into the underlying patterns and factors contributing to stock market volatility, ultimately contributing to a better understanding of financial markets.**

***Index Terms*—Stock Market, Machine Learning, Circuit Breakers, Stock Volatility**

I. Introduction

Stock markets have always been profitable investment options with excellent potential for wealth generation. However, stock markets are also volatile, and can experience sudden and dramatic price swings. These price swings can be caused by a variety of factors, including changes in economic conditions, news events, and investor sentiment. When stock prices experience a sudden and substantial decline, it can lead to panic selling. Panic selling is when investors sell stocks out of fear, without considering the underlying fundamentals of the company. Panic selling can quickly accelerate the decline in stock prices, leading to a market crash. The stock crashes can dislocate economic activities and disastrously affect the investors whose income and wealth are based on these financial assets. Circuit breakers are administered to particular stocks or indices to curb the extreme price fluctuations that occur due to panic selling or buying of stocks. Circuit breakers are a critical tool in managing volatility in the stock market.

The first circuit breaker system was introduced in the United States after the "Black Monday" stock market crash of October 1987. Black Monday was a day when the Dow Jones Industrial Average fell by over 22%, its largest single-day decline in history. The circuit breaker system was credited with helping to prevent a more severe crash. Since then, circuit breaker systems have been implemented in stock markets around the world. These systems have helped to prevent several major market crashes, including the 2008 financial crisis and the 2020 COVID-19 pandemic.

By halting trading when there is a sudden and significant price movement, the system gives investors a chance to assess the situation and make rational decisions before trading resumes. The circuit breaker system is not without its critics. Some argue that it can actually exacerbate market volatility by causing investors to panic and sell stocks when trading is halted.

Here are some of the drawbacks of circuit breakers:

- They can prevent investors from taking advantage of opportunities during periods of volatility.
- They can be triggered by false or misleading information.
- Traders can manipulate circuit breakers to their advantage.

Ultimately, the decision of whether or not to use circuit breakers is a complex one. There are both pros and cons to consider, and the best decision may vary depending on the specific circumstances.

II. Problem Statement

The challenge addressed by this project is the persistent ambiguity surrounding stock market volatility and the effectiveness of circuit breakers in managing losses during periods of unpredictable price fluctuations. Despite the successful development of a preliminary algorithm to identify circuit breaker instances, there remains a critical gap in understanding the intricate dynamics governing market volatility. This phase aims to address this gap by deploying a diverse set of machine learning algorithms, including Random Forest, Gradient Boosting, Prophet, and LSTM. The emphasis lies on fine-tuning hyperparameters and rigorously assessing the models' performance to unravel the hidden patterns and contributing factors to stock market volatility. The overarching problem is to enhance our comprehension of financial markets, providing valuable insights that can guide investors, regulators, and researchers in navigating the complexities of market behavior and risk management.

III. Modelling and Setbacks

In the initial stage of the project, the circuit detection algorithm exhibited a high accuracy in identifying circuits through manual verification. The algorithm currently achieves a precision of around 0.9 and an accuracy of 0.83, serving as our benchmark for improvement. Throughout this project phase, we've explored the implementation of diverse machine learning models, employing distinct features and datasets for each model iteration.

1. Logitstic Regression:
Our initial approach involved employing Logistic Regression to forecast the occurrence of a circuit on the next day, utilizing two months' worth of historical data. The dataset comprised features such as Opening Price, Closing Price, Volume, Adjusted Closing, High, and Low. To enhance the model, we introduced an additional feature named 'per_diff,' which calculated the percentage difference between the closing price of a given day and the subsequent day. The target variable was then derived based on this difference, deviating from the algorithm developed in the previous phase of the project. This adjustment was made as a preliminary step to gauge the suitability of Logistic Regression for our dataset. However, the model exhibited an exceptionally low accuracy of 17.68%, leading us to reject its viability for our predictive purposes.

2. Random Forest:
Utilizing the same set of features mentioned earlier, we sought to implement a classification approach using Random Forest. Random Forest, known for its ability to capture intricate relationships in data, operates as an ensemble of decision trees. Each tree is trained on a subset of the data, contributing to the model's flexibility and robustness. Given its capacity to handle non-linear relationships and interactions between variables, Random Forest appeared promising for analyzing stock price data. However, our findings indicated that the accuracy of the model did not show substantial improvement. This outcome can be attributed to the inherent challenges associated with stock price data, characterized by noise and abrupt changes influenced by various factors such as market sentiment, news, and economic events. The movements in stock prices often exhibit non-linear patterns and complex dynamics. While Random Forest excels at capturing non-linear relationships, its performance may still be hindered by the intricate and ever-changing nature of stock markets.

3. Using Fundamental Ratios:
The training data used in the previous models has a high degree of multicollinearity, meaning that the features are strongly correlated. To address this, we conducted additional feature engineering. We explored factors beyond stock prices that might help predict circuit breakers. Investors and analysts often rely on fundamental financial ratios to assess a stock's potential. These ratios can influence trading and contribute to fluctuations in stock prices, potentially triggering circuit price limits.

For example, a higher price-to-earnings (P/E) ratio compared to competitors may suggest an overvalued stock due for a correction. This perception could prompt investors to sell, causing a decline in stock prices. Changes in other fundamental ratios, such as the debt-to-equity ratio, return on equity (ROE), or earnings per share (EPS), can also impact stock trading. These ratios signal shifts in a company's financial position or performance, influencing investor confidence and trading activity.

Fundamental financial ratios are crucial in stock analysis and trading. Fluctuations in these ratios can significantly affect a stock's performance. However, the dataset for these ratios is not available, requiring the generation of data using accessible parameters. Thus far, we have calculated EPS, P/E ratio, P/S ratio, P/B ratio, and P/CF ratio for some stocks. Unfortunately, these derived attributes still exhibit multicollinearity, making this dataset unsuitable for training machine learning models to predict circuit breakers.

4. Prophet Model:
Due to issues like class imbalance and multicollinearity, we decided to explore the Prophet model for analyzing trends and seasonality in stock price data. This approach allows us to predict future stock prices, aiding in forecasting potential occurrences circuit breakers.

Prophet is an open-source forecasting tool developed by Facebook. It is designed to handle time series data for forecasting purposes, and it is particularly well-suited for datasets with strong seasonal patterns and multiple seasonality. Prophet is widely used for business applications, such as sales forecasting, demand planning, and financial market prediction. Prophet decomposes the time series into three main components: trend, seasonality (daily, weekly, and yearly), and holidays. It accommodates both yearly seasonality and weekly seasonality, allowing the model to capture complex patterns in the data. It provides uncertainty intervals for the forecast, helping users understand the confidence level associated with the predictions.

The results of implementation of Prophet model on SALASAR stock are provided further. The sole feature utilized in this analysis is the Closing Price of the stock over two years. We configured the 'daily_seasonality' hyperparameter as 'True', enabling the model to account for daily patterns in the time series data.

Figure 1 breaks down the distinct components of the stock's closing price. It is evident that the stock price exhibits an upward trend over the two-year period. Examining the weekly component, it becomes apparent that prices typically decrease on Mondays when the market reopens after the weekend, experience an increase on Tuesdays, and then fluctuate within a similar range for the remainder of the week.

Figure 2 illustrates the variance between predicted values and actual values of the stock price. It's evident that the model struggles to capture short-term trends in the data. Despite a

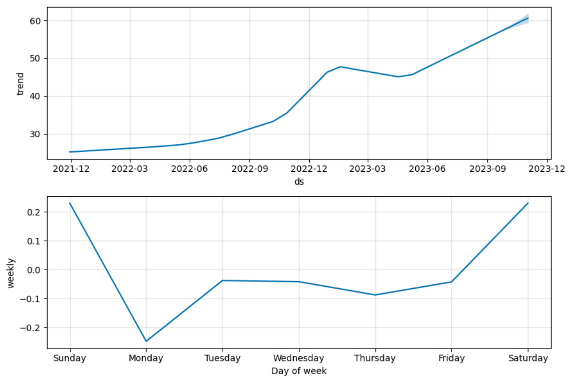

Fig. 1. Trends Observed in SALASAR Stock

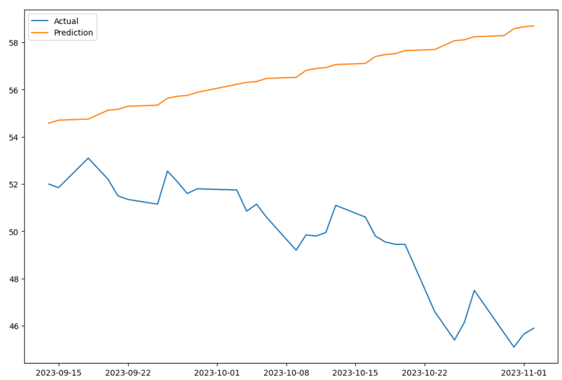

Fig. 2. Actual vs Predicted Price for SALASAR Stock using P

mean absolute percentage error (MAPE) of 14.29% for this specific stock, which is deemed acceptable in some instances, the model's limitations in capturing short-term trends are noticeable.

Table I displays the performance metrics following cross-validation in the Prophet Model for the following values of the hyperparameters:

- initial='365 days'. Specifies the initial training period for the model. In this case, the model is trained on the first 365 days of the time series.
- period='100 days'.Defines the length of each training period. After the initial training, the model is retrained every 100 days on the updated historical data.
- horizon = '30 days'. Sets the forecast horizon, which is the number of days into the future for which predictions will be made in each iteration. In this case, predictions are made for the next 30 days.

MAPE is the mean absolute percentage error calculated for the particular forecast horizon. The column Coverage represents the fraction of actual values that fall within the uncertainty interval.

Certainly, as the forecast horizon extends, there is a noticeable increase in the Mean Absolute Percentage Error (MAPE) and a decrease in coverage. In other words, the confidence intervals fail to encompass the actual values. This observation strongly suggests that the model is ill-suited for making accurate predictions for events further into the future.

TABLE I
PROPHET PERFORMANCE METRICS

Horizon	MAPE	Coverage
5 days	0.102	0.67
10 days	0.068	0.83
15 days	0.181	0.33
20 days	0.235	0.11
25 days	0.125	0.42
30 days	0.191	0.33

5. Long Short-Term Memory:

Recognizing that machine learning models might not be suitable for achieving our desired objective, we have shift our focus to Deep Learning methods. Our initial approach involves using Long Short-Term Memory (LSTM) networks to predict stock prices and eventually circuit-breakers.

Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) architecture that is particularly well-suited for sequence prediction tasks, including time series forecasting such as predicting stock prices. LSTMs are designed to capture and learn patterns in sequences, making them effective for modelling temporal dependencies in time series data. An LSTM network consists of memory cells that can store and retrieve information over long sequences. The architecture includes three gates: the input gate, forget gate, and output gate. These gates control the flow of information and help the network learn which information to remember and which to forget.

Once again, we are exclusively utilizing the closing price of a stock over a one-year period, and we scale it using the MinMaxScaler. We have tested two different models: one being Simple RNN and the other being LSTM. Simple RNN model's architecture is illustrated in Figure 3. The stock example used here is TATASTEEL.

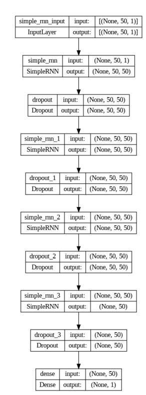

Fig. 3. Simple RNN Model Architecture

Figure 4 showcases the comparison between actual values and predicted values using the simple RNN model. Notably, deep learning methods exhibit a better capability to capture short-term dependencies compared to the previously employed model. After 50 epochs, the Mean Squared Error (MSE) loss has decreased from the initial value of 0.7816 to 0.1151. This reduction in loss indicates an improvement in the model's ability to minimize the difference between predicted and actual values during the training process.

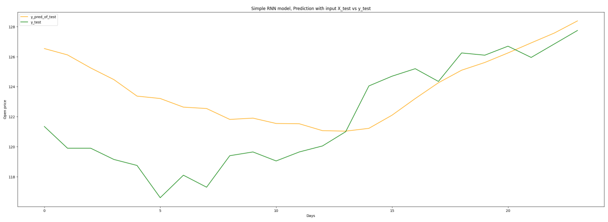

Fig. 4. Simple RNN Model Actual vs Predicted Values

In our experimentation, we explored an alternative architecture using LSTM, as depicted in Figure 5, and compared actual values to predictions in Figure 6. The initial loss for this model was 0.0556, reducing to 0.0137 by the tenth epoch. Once again, we observe that deep learning methods outperform our previous approaches with better performance on our data. While these predicted prices can aid in foreseeing future occurrences of circuits, our focus in the next phase of the project will be on building a model that can further decrease the loss for more accurate predictions.

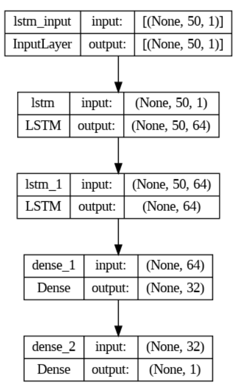

Fig. 5. LSTM Model Architecture

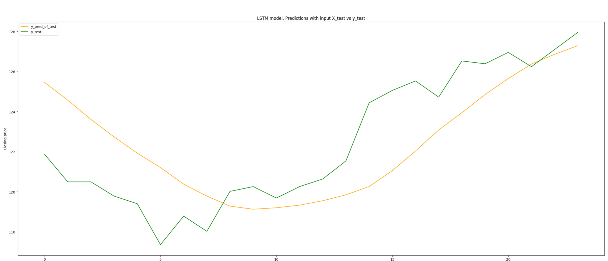

Fig. 6. LSTM Model Actual vs Predicted Values

IV. Conclusion and Future Work

In this phase, we've demonstrated that traditional machine learning methods are not well-suited for handling volatile datasets like those in the stock market. Deep learning techniques, particularly when dealing with sequential data, have shown superior performance. Looking ahead to the next phase of the project, our goal is to implement multivariate time series classification for predicting the occurrence of circuit breakers. This involves incorporating additional features through further feature engineering, leveraging the insights gained from the current results, and employing advanced deep learning models to enhance predictive accuracy.