# Stock Sentiment Analysis using LSTM

Adesh Gupta
IIT Roorkee

June 2024

**Abstract**

This report describes my approach towards the problem statement of Stock prediction using Sentiment Analysis of news reports. This approach uses LSTM regression model for predicting the Close price and buy sell signals are generated according to the predicted Close price data. Then a trading algorithm is built around the generated signals and thorough evaluation is done.

## 1 Introduction

The goal of the project was to develop a pipeline that predicts the stock price movements using sentiment analysis over the news headlines scraped from the web. The proposed method uses LSTM model which is considered great for series data which we are dealing here with. Using LSTM leads to improved Sharpe ratio of the trading algorithm and significant returns are observed. The Sentiment Analysis of headlines is done using the FinBert Pipeline [1] and creating features around it to feed into the model.

## 2 Data Scraping

Market Insider is used as the main source for getting the headlines data. The choice of website was clear due to the fact of clearly structured html and data segregated in pages. So data of long back ago can be requested without any effort to create a crawler or a bot. Stock trend data is obtained through Yahoo Finance API. The data between 2020 -2024 is chosen as the ideal data to work with due to consistent number of headlines per day available in this period for the chosen stocks.

## 3 Data Analysis and cleaning

Dividends and Stock Splits played no role in our analysis hence these columns were dropped. The data between 2020 -2024 is chosen as the ideal data to work with due to consistent number of headlines per day available in this period for the chosen stocks. We also need to consider the fact that there are some sparse dates for which no data are available so we can't judge for those days. The train data was all the dates between 2020-2023 and all the data of 2023 and after was used as test data.

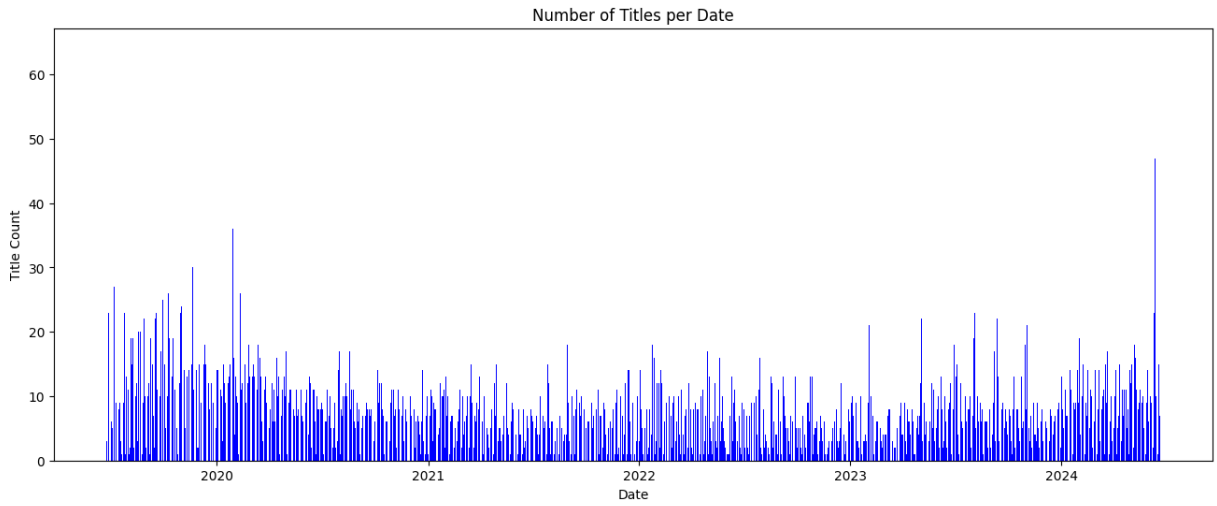



Figure 1: Good density of tiles per day between 2020-2024 period

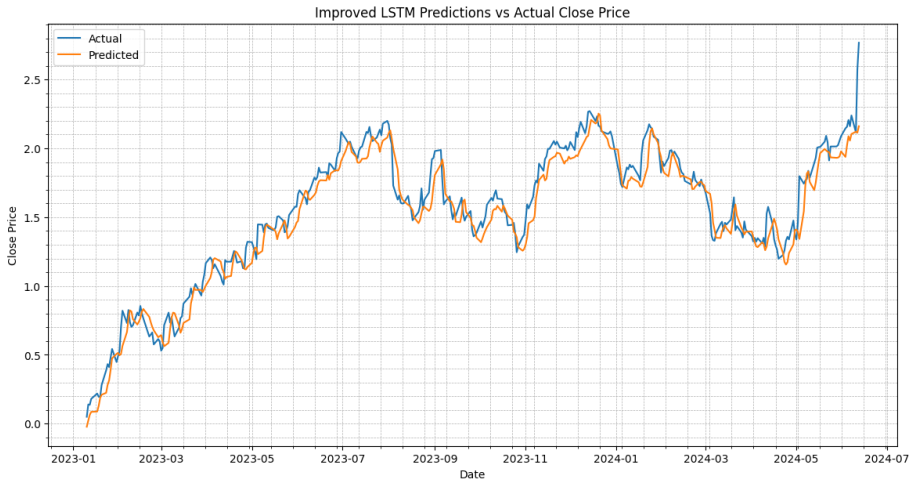



Figure 2: Predicted vs Actual Close on AAPL stock

# 4 Sentiment Analysis

Pre-trained Finbert model from hugging face was used for labelling each headline with its sentiment type and sentiment score. A feature is defined for each headline which is product of the weight defined for its sentiment type and its sentiment score. The weights were initialized as follows:

- Positive: +1
- Negative: -1
- Neutral: 0.1

Then all the headlines were grouped by the dates and the mean and median of the sentiment feature is calculated for each day. These are the extracted features per day that will be fed in the LSTM model for making predictions of the Close price.

# 5 LSTM Regression

A simple LSTM model was used. First the data was converted to sequences of length 5 and these were used to predict the Close price for the next day. LSTM model was chosen because it gave best results and generated best signals for a trading algorithm. Input features were: Close, Open, High, Low, Volume, mean feature, median feature for past 5 days as series. And output was expected to be the Close of the next day. The data was scaled using Standard Scaler for efficient learning process. Two LSTM layers of 100 were used with Dropout of 0.2 to avoid overfitting. Standard Adam optimizer was used for the model with mean square error as loss. The training was done over 50 epochs with a validation split of 0.2. This predicted the Close price of the stock very well, evaluation of which is as follows:

# 6 Generating Buy Sell Signals

The trading part is kept simple and only two types of signals are generated. 1 for a buy signal and 0 for a sell signal. The signals are generated using the percentage change in the predicted close price of two consecutive days. If it is positive then signal generated is 1 (means one should buy the stock at that time). If return is negative the generated signal is 0 (means one should sell the stock).

# 7 Trading Algorithm

The trading algorithm is kept simple. Starting with a value of $100,000, the algorithm buys the maximum number of stocks when a buy signal is received and sells all stocks on a sell signal.

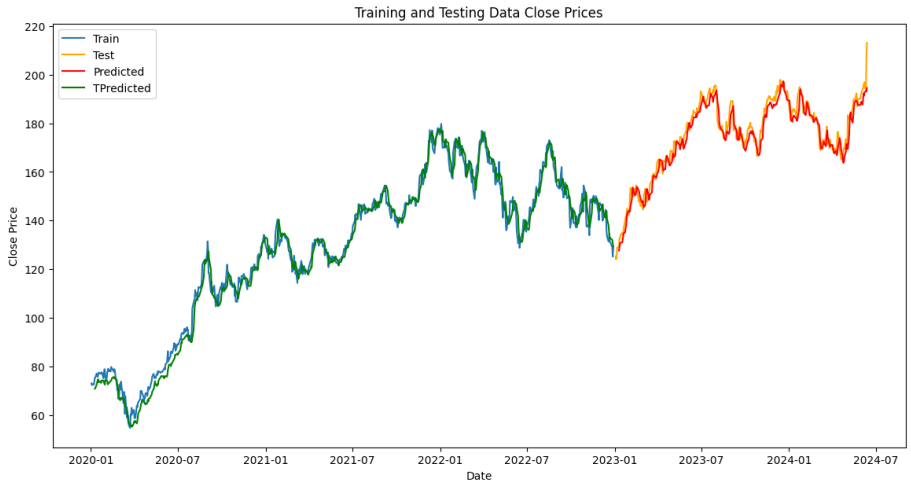
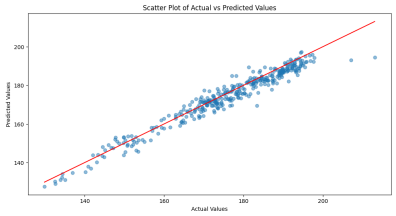
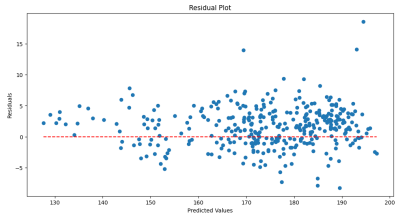


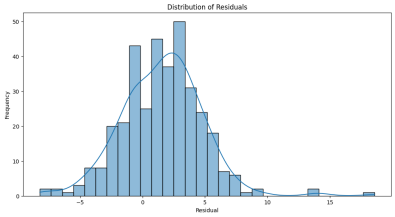

Figure 3: Visualizing predicted vs Actual Close data on train and test parts




(a) Scatterplot b/w actual and predicted price




(b) Plot of residuals




(c) Residuals following normal distribution

Figure 4: Data visualizations of prediction

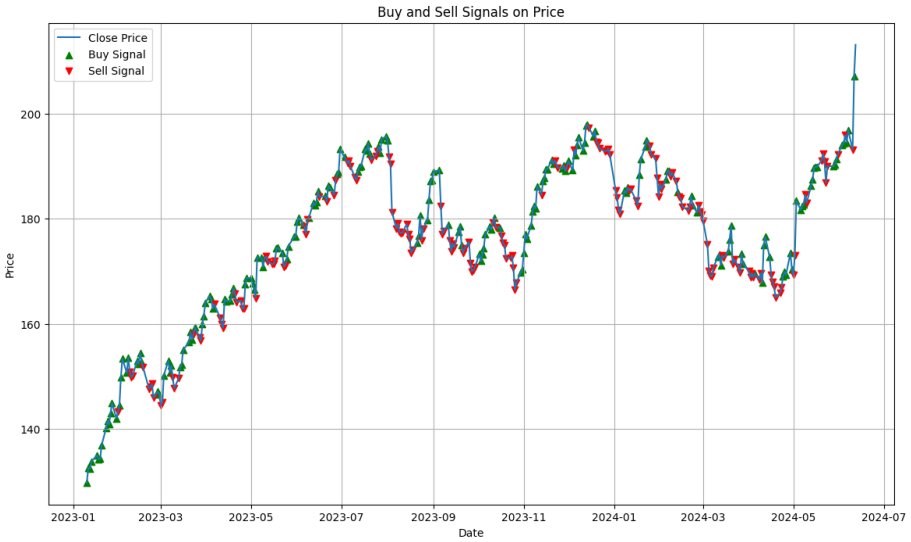



Figure 5: Generated buy (green) and sell (red) signals over the test data (AAPL)

# 8 Discussion

## 8.1 The results

The evaluation metrics of the regression model came out to be as follows:

- Mean Absolute Error (MAE): 0.0920
- Mean Squared Error (MSE): 0.0138
- Root Mean Squared Error (RMSE): 0.1175
- R-squared ($R^2$): 0.9431

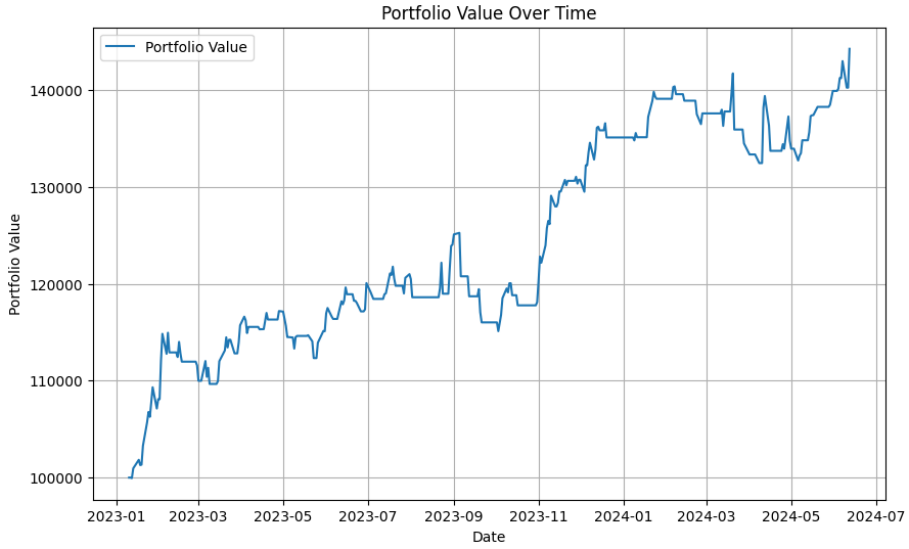



Figure 6: Portfolio returns over time period of a year on AAPL

These are the values after the scaling of data and its prediction. So R squared can be considered as much better metric than all of the else as it tells about how better the data has fitted and does not depends on the scale of the data.

The evaluation metrics of the trading algorithm came out to be as follows:

- Sharpe Ratio: 1.8151
- Maximum Draw down: -0.0810
- Number of Trades Executed: 97
- Win Ratio: 0.3202

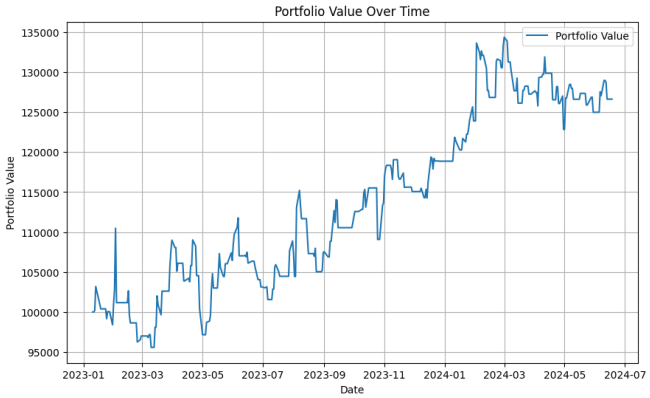



Figure 7: Portfolio using same pipeline on AMZN, Sharpe Ratio: 0.77

### 8.2 Area of improvement

There are several areas of improvement for the current algorithm. More text can be extracted by going on each individual link and iterating through it to gather more textual data and making the model much more reliable. Other text pre-processing techniques can be used. A more complex LSTM architecture can be used to fine tune the predictions. The trading algorithm can be improved by using risk factor, stop loss, and other trading strategies to maximize the profit.

## References

[1] Dogu Araci. Finbert: Financial sentiment analysis with pre-trained language models, 2019.