Sentiment Analysis on r/WallStreetBets and predicting the success of their Market Strategies

CSE342 - Statistical Machine Learning Course Project

Ritika Thakur
2022408
CSAI

Saksham Singh
2022434
CSE

Fig. 1. r/WallStreetBets

***Abstract*—This project aims to build a simple ML model that allows users to judge whether the market decisions in r/WallStreetBets - the biggest Reddit community for stock trading - are safe for an average person to invest in, and thus automates this process.**

I. Problem Statement and Motivation

In recent years, social media platforms such as Reddit's r/WallStreetBets have emerged as influential forums for discussions and opinions on stock investments. However, analyzing the vast amount of unstructured data generated by these communities, not to mention their colorful and profane jargon, and aggressive trading strategies, it presents a significant challenge for investors seeking to make informed decisions. The aim of this project is to develop an AI-driven solution that leverages sentiment analysis on r/WallStreetBets posts to identify potentially lucrative investment opportunities and subsequently predict the future value of these stocks.

A. Sentiment Analysis

Develop a sentiment analysis model capable of accurately assessing the sentiment expressed in posts and comments on r/WallStreetBets. The model should be trained to differentiate between positive, negative, and neutral sentiments regarding specific stocks mentioned in the forum.

B. Stock Prediction Model

Design and implement a machine learning model for predicting the future value of stocks based on historical price data, market indicators, and sentiment analysis results. The model should be capable of generating accurate predictions over various time horizons.

C. Integration and Deployment

Integrate the sentiment analysis and stock prediction models into a unified system capable of processing real-time data from r/WallStreetBets and generating investment recommendations. Deploy the system in a scalable and accessible manner, ensuring seamless operation and user interaction.

In this deadline we aim to show the results of our Stock Prediction Model.

II. Dataset Details

Instead of using Quandl as proposed in our initial project proposal, we found out Yahoo Finance's dataset and the respective YFinance python library to be better suited for our needs but Quandl can be used too for the same with some minor changes.
Yahoo Finance hosts the data for every major listing on Nasdaq and provides the open price, the high, the low, and the close price of every single trade day for that listing since the documented history and to the present date.
We intend on using the close price variable to predict the values, thus making our model univariate. For the demonstration, we have run our model on the stock history of Apple and Microsoft and their stock trends from 1st January 2016 to 1st January 2024. We have divided the dataset into 2 parts - namely train (65%) and test (35%).

III. Proposed Architecture

We intend to use Long Short Term Memory Neural Networks (LSTM) for our architecture - majorly because of its ability to learn from sequential data while considering long-term dependencies. In the context of stock prediction, past stock prices and other relevant factors form a time series, and LSTM can effectively capture patterns and trends within this sequential data. Also their ability to handle non-linear data and memory retention, they seem to be the perfect model for our application.
We have used Tensorflow's Sequential model with LSTM and Dense layers for libraries. The model trains and predicts for each day using the trends of the last 100 days to make its decisions. We have used 3 layers for LSTM and one layer

for Dense, and have computed our losses in terms of mean-squared error.

IV. Visualisations

The below are our findings when running the model on the Apple's stock history:

A. Apple

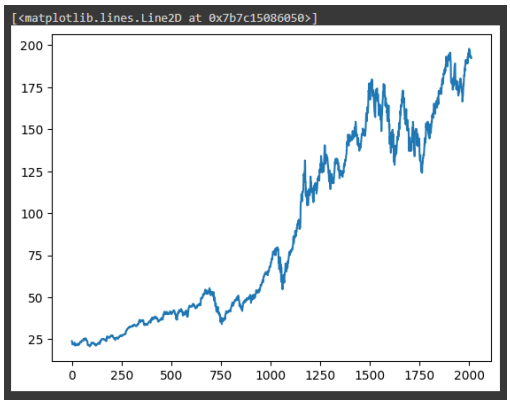

Fig. 2. Apple Dataset

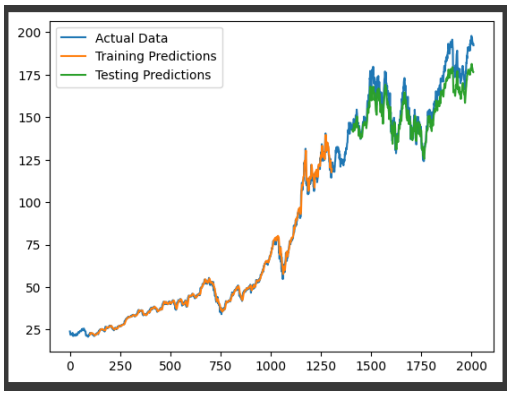

Fig. 3. Predictions on Train and Test set on the Apple Dataset

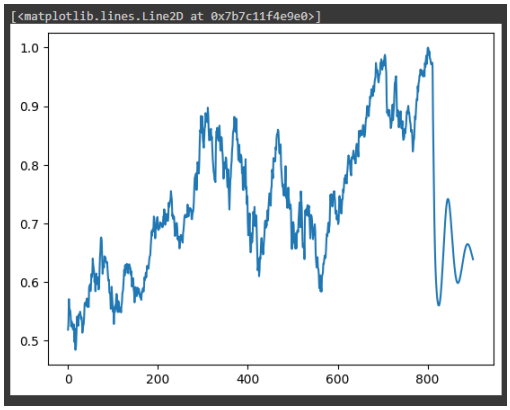

Fig. 4. Apple Predictions for next 90 days

B. Microsoft

The below are our findings when running the model on the Microsoft's stock history:

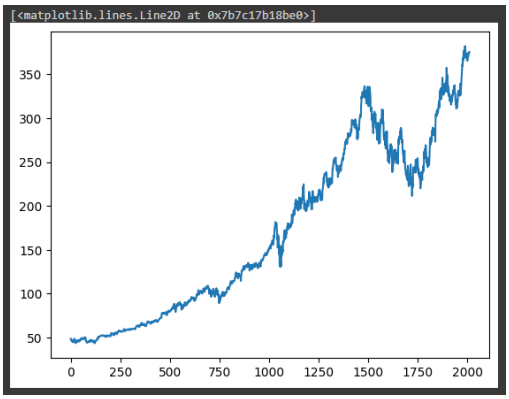

Fig. 5. Microsoft Dataset

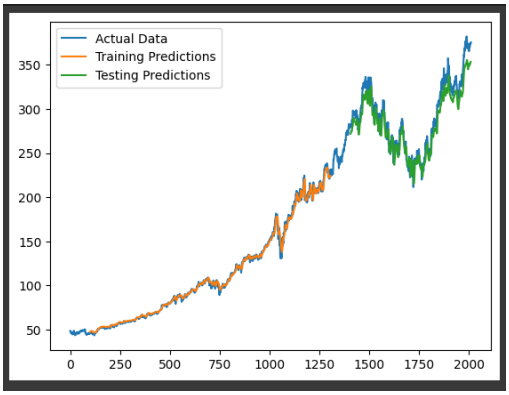

Fig. 6. Predictions on Train and Test set on the Microsoft Dataset

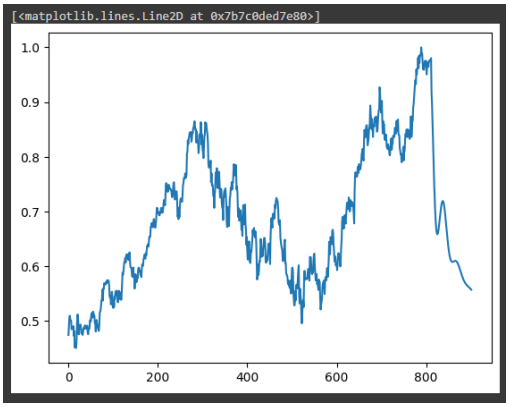

Fig. 7. Microsoft Predictions for next 90 days

V. Results

A. Apple

- MSE on the train and test set at the last epoch of the LSTM model for each data point (mean) were 9.9508e-05 and 0.0019 respectively.

- The net MSE on train set and test set were found out to be 62.112 and 154.387 respectively.
- The future predictions seem a little flawed considering the huge dips and rises in our predictions compared to the past trends.

B. Microsoft

- MSE on the train and test set at the last epoch of the LSTM model for each data point (mean) were 9.5480e-05 and 0.0012 respectively.
- The net MSE on train set and test set were found out to be 125.20580 and 284.346768 respectively.
- The future predictions seem a little flawed considering the huge dips and rises in our predictions compared to the past trends.

VI. Analysis of Result

- LSTM proved to be a good regressor for something like stock prediction which is factored by a lot of real world variables which one might say to be random, but using the past data it was able to give a good estimate.
- Our model fails to predict for large durations in the future because of its data being referenced from another future predicted data. At best, it can give a prediction for the next day or two.

VII. Individual contributions of each group partner

Everything from discussion, implementation and report writing was a collective effort and done together.

VIII. References

- Greg Hogg's youtube video on LSTMs and Stock prediction model - https://www.youtube.com/watch?v=CbTU92pbDKw
- Krish Naik's youtube video on LSTMs and Stock prediction model - https://www.youtube.com/watch?v=H6du_pfuznE