

Stock Price Prediction

Description

Hedge funds use a lot of past data to understand and predict the price of the stock in future and base their decisions on the predictions. The price of the stock varies based on how the market thinks the company is going to perform given the current economic and company's financial conditions. Based on these conditions, buyers and sellers place their stocks at a particular price and the stock price is determined based on the last price at which the transaction has taken place. Many hedge funds try to use ML based approaches to predict the price of the stock on the next day and put a buy or sell strategy to get maximum profit. In the capstone project we will create a ML model to do the same.

There has been a lot of research done in the past on the topic as well

(http://www.ijirccce.com/upload/2016/november/153_Prediction.pdf,

<http://web.cecs.pdx.edu/~edam/Reports/2004/Garner.pdf>

). Many papers have tried to use sentiment analysis to predict the stock price

(<https://link.springer.com/article/10.1007/s00521-019-04504-2>) however we will just focus on

the OCHL data for the stock. I would personally try to build a stock price ticker to understand how well a machine learning model predict a price and if I can personally use it for my investment purposes.

Problem Statement

In this project our aim is to create a model that can predict if we should buy or sell the stock at the opening price today so that we can make the profit at the end of the day. Thus, the aim of the project is given the past data and the opening price today, can we predict if the closing price be greater or less than the opening price today.

The solution will be a simple web app that can predict if the closing price is greater or less than the opening price. I will be doing this for only a select tickers (AAPL) to begin with and expand the site to more tickers later on.

Datasets

I will be creating the data using the free open source APIs mentioned in the previous stock prediction notebook

(<https://docs.google.com/document/d/1ycGeb1QYKATG6jvz74SAMqxrlek9Ed4RYrzWNhWS-0Q/pub>)

- [Yahoo! Finance](#): You can directly query for a stock through the web API, or download a dump of .csv files and use them.
- [Bloomberg API](#): Multiple APIs available, including Python.
- [Quandl](#): Also multiple APIs, including Python.

I will download the csv dump for training the data and will try to extract the stock price using the API for the stock to predict if the stock should be a good buy for today or not.

Solution Statement

I will be using LSTM model to predict boolean (greater or less than) for the AAPL stock ticker. The input data will consist of the OPEN, CLOSE, VOLUME, HIGH, LOW of previous day and the OPEN of the current day. The output will be a Boolean value which will tell if the stock is going to be higher or lower than the open price today.

Benchmark

We will compare our model based on the accuracy of the model.

If we randomly choose that the stock will go up or down based on the stock price, then we will get 50% accuracy. That is the benchmark I am considering. The model should perform significantly better than the benchmark model for us to make profit in the stock markets.

I will also compare the model with a simple XGboost based model to understand the performance gains of using an LSTM based approach over a simpler Xgboost based approach.

Evaluation Metrics

For evaluation I will be using accuracy of the model, precision, recall and ROC AUC of the model. These measures should tell us in general how the model is doing overall and on different thresholds. Generally, if the model with high threshold values should have high profit margins as we are taking less risk in the stock market.

<https://towardsdatascience.com/predicting-the-stock-market-with-machine-learning-benchmarking-44181286389>

Project Design

These are the list of tasks I have in mind to implement a stock price predictor:

1. Gather the required data using the APIs mentioned above
2. Split the data into train, validation and test data.
3. Perform data analysis on the train data and extract relevant features. Some features I can think of now are:
 - a. Moving average
 - b. Volatility
4. Train the LSTM and XgBoost model using the above data
5. Perform analysis on the best model using evaluation criterion mentioned above to deploy in production
6. Create a web site to query the model and display the results if the stock price will be higher or lower than the current stock price.

Learning Resources

As mentioned in the original stock price prediction notebook:

- [Machine Learning for Trading](#), Tucker Balch (Georgia Tech and Udacity)
- [Stocks and Bonds](#), Khan Academy