Udacity Machine Learning Nanodegree 2020

Capstone Proposal

Stock Prediction Using Deep Learning

Nitu Nivedita

June 2020

1 Domain Background

This project aims at using Deep Learning and Recurrent Neural Networks in order to predict stock prices. Accurate stock market prediction is of great interest to investors; however, stock markets are driven by volatile factors among myriad other factors. For achieving maximum prediction accuracy, this Stock Prediction model tries to use deep learning and time series modelling methodologies that can look at the history of a sequence of data and correctly predict what the future elements are going to be.

Since 1600s, the ultimate goal of any investor has been to make profit. Being able to predict which assets will appreciate or depreciate in value over time is of great interest to one and all. Individual investors, hedge fund firms or wealth management companies, have all been seeking to understand and accurately predict the stock market in order to make profit.

By using neural networks concepts and time-series - Long Short-Term Memory, LSTM – this project aims to build models that are powerful, especially for retaining a long-term memory, by design and that can predict stock prices with high confidence.

This project uses the concepts of Recurrent Neural Networks published in the research paper Comparative Study of Stock Trend Prediction using Time Delay, Recurrent and Probabilistic Neural Networks [1] D. C. Wunsch et al., "Comparative Study of Stock Trend Prediction using Time Delay, Recurrent and Probabilistic Neural Networks," IEEE Transactions on Neural Networks, Institute of Electrical and Electronics Engineers (IEEE) as well as the more recent [2] An ensemble of LSTM neural networks for high-frequency stock market classification long–short-term memory (LSTM) neural networks for stock prediction.

My motivation in solving this problem is both a) to understand the dynamics of the stock market better and b) to use the power of Machine Learning to discover buying opportunities. My goal is to develop a structured LSTM network for time series predictive modeling and use the LSTM model to predict stock price and stock price movement for taking Buy/Sell/Hold decisions.

2. Problem Statement

Predict the future price of a given stock, given the historical performance of the said stock and a Buy/Sell/Hold recommendation decision

Accurately predicting Stock Prices has been a long-standing problem that mankind has tried to solve. While there is no model that exists, which can accurately and with complete certainty predict all stock prices accurately, yet this is one of the most interesting problems that can be dealt with by using Machine Learning and Neural Networks. Given the past historical

performance of stocks, this project tries to solve of the problem of predicting the stock prices in the market as accurately as possible. The problem is replicable and quantifiable.

3. Datasets and Inputs

The datasets and inputs used in this project will be from Alpha Vantage and S&P 500 Full dataset with 10 Years of Open/Close/Low/High/Volume data.

(1) Yahoo Finance – To get the historical and latest stock prices. Using pandas-datareader (a pandas extension with built-in web-scraping features) , we will scrape the stock prices of last 10 years

(2) Alpha Vantage – At the outset, an API key will be required, which we will be obtained for free at the above link. After that, the API key will be assigned to the api_key variable.

Alpha Vantage APIs are grouped into four categories: (1) Stock Time Series Data, (2) Physical and Digital/Crypto Currencies (e.g., Bitcoin), (3) Technical Indicators, and (4) Sector Performances. All APIs are real time: the latest data points are derived from the current trading day. This project will be using the Stock Time Series Data.

(3) S&P 500 Full dataset - Data will be downloaded for S&P 500 stocks into a "Stocks" folder. Later this data will also be uploaded in a folder in S3 bucket which the model can use.

For each stock, the inputs will contain multiple metrics - opening price (Open), highest price the stock traded at (High), how many stocks were traded (Volume) and closing price adjusted for stock splits and dividends (Adjusted Close) .

Characteristics of input datasets :

- Open: Opening stock price of the day
- Close: Closing stock price of the day
- High: Highest stock price of the data
- Low: Lowest stock price of the day
- Volume : Volume of shares tracded on a day
- Adjusted Close : the closing price that has been amended to include any distributions and corporate actions that occurred at any time prior to the next day's open
- company_name

Since this goal of this project is to predict future stock prices through a Long Short-Term Memory (LSTM) method, it is important to understand the historical performance of the stocks, the pattern and risk factors and then predict the stock movement in the future. Hence, the above two datasets will provide ample data points we require to train, test and evaluate the model

4. Solution Statement

To solve this problem, I will use recurrent neural network. I will be using Long Short-Term Memory as the model - LSTMs are an enhanced version of recurrent neural networks (RNNs). LSTMs are a type of RNN that remember information over long periods of time, making them better suited for predicting stock prices. The stock that the user selects will be the target. The model will be trained and used to predict the closing price stock price of the chosen stock using LSTM.

The solution will be divided into the following parts :
1) Determine change in price and volume traded of the stocks , over time

3) Simple Moving Average of the input stocks

4) Determine correlation between different stocks – This is done to find clusters of stocks that a ppreciate or decline together or have similar market factors that impact them

5) Determine risk factors by analyzing average daily returns of the stocks and correlation. This is to determine the risk of the portfolio

6) Predict closing stock price and stock price movement of any given stock.

7) Provide Buy/Sell recommendations. Different classification machine learning approaches will be used to determine Sell/Buy/Hold recommendations

5. Benchmark Model

The benchmark model for this project would be using linear regressions since my main goal to compare and contrast the accuracies of the various machine learning models. This benchmark will use exact the same input as our LSTM network model and provide a benchmark performance for the LSTM.

6. Evaluation Metrics

1) For evaluation of the model, RMSE will be used. Root mean squared error (RMSE) can provide what is the average deviation of the prediction from the true value, and it can be compared with the mean of the true value to see whether the deviation is large or small. We can plot the predicted and actual data. Valid and predicted prices visualized can be used to evaluate if the model passed.

2) To analyze our model's effectiveness we can validate our model against untested data. One metric we can create is a prediction interval (similar to a confidence interval) which will tell us with some high degree of confidence in the true range of our prediction. We can use the root mean squared error for getting to a prediction interval

3) I will use R-square and RMSE to check the model performance, on both benchmark model and LSTM model

4) I will use scikit-learn LinearRegression and use Keras's LSTM function to build the neural networks. The networks will have two or three layers

7. Project Design

Data Cleansing

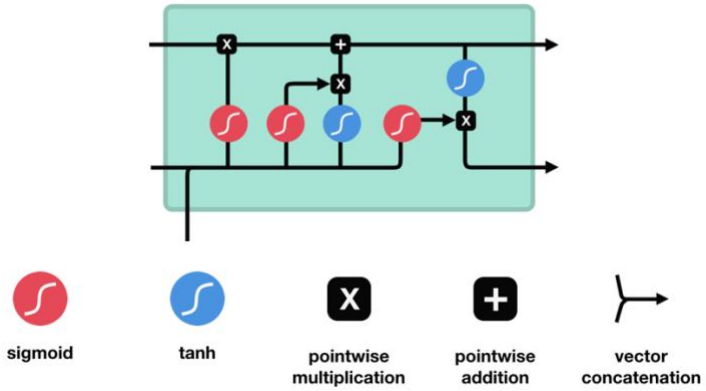
a) Scrape the data set from Yahoo Finance and cleanse the acquired data.
b) We have prices of different stocks, they are in different scales and hence for the neural network to converge, we need to scale.
c) Normalize the data for the LSTM networks which will be done using the function MinMaxScaler of the sklearn library. Create the scaled training data set
d) Split the dataset into the training (80%) and test (20%) datasets

Build Design

a) The First step is to import the libraries needed - sklearn library, Keras, seaborn and matplotlib for visualization

b) Build the LSTM model, compile and train

LSTM has a similar control flow as a recurrent neural network. It processes data passing on information as it propagates forward. The differences are the operations within the LSTM's cells. These operations are used to allow the LSTM to keep or forget information.



The core concept of LSTM's are the cell state, and it's various gates. The cell state act as a transport highway that transfers relative information all the way down the sequence chain. As the cell state goes on its journey, information gets added or removed to the cell state via gates.

c) Get the models predicted price values
d) A vast set of indicators from the finta.py library will be used to make several complex moving averages, oscillators, strength indexes etc

Implementation Steps

1. This model will be implemented using Keras module.
2. Create an object for the model with Sequential function.
3. Add layers to the model
4. In this network first a LSTM layer will be added which takes the 3 dimensional array as input and has dimension 50 i.e the number of neurons.
5. Then we compile the function using a loss function parameter, the optimizer and the metrics we want to use to check the model's efficiency.
6. Call function fit to train the model
7. Use the model to predict closing price of any given stock (ex. AMZN, AAPL, GOOG etc)

8. Presentation

Results will be presented/plotted on a graph showing visual trends. If time permits, I will add a Lambda and API Gateway front-end for visualization

References :

https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21