



Machine Learning Engineer Nanodegree Capstone Proposal

20.1.2020

Nishit Singh

Domain Background

Besides the insurmountable contributions of Machine Learning in numerous vital fields, the stock prediction has also seen rapid growth. Over the years, several Machine Learning techniques such as the random trees and multi-layered perceptron have been employed to improve efficiency in the prediction of highly varying stock data. There are several factors involved in stock prediction - physical and psychological factors, rational and irrational behaviors, etc. All these aspects combine to make stock prices volatile and very difficult to predict. The current trend is to use LSTMs (Long Short-term Memory) networks for stock prediction. LSTMs are very powerful in sequence prediction problems because they are able to store past information, which is an important factor in stock prediction. After a thorough evaluation, I have decided to use LSTMs for Time Series Prediction on the **nifty50 stock market data** (2000-2019)

(<https://www.kaggle.com/rohanrao/nifty50-stock-market-data>)

Problem Statement

The purpose of this project is to integrate the best performing Machine Learning Technique in a stock price prediction algorithm which Utilizes the **nifty50 stock market data**

(2000-2019) (as referenced above). The algorithm will take the stock name and date of prediction as input from the user and give the most optimal output.



Datasets And Inputs

In this project, I will use all the stocks in the **nifty50 stock market data**

(<https://www.kaggle.com/rohanrao/nifty50-stock-market-data>) which I got by using Google Dataset search, as inputs and targets. For each stock, the data will contain Open, High, Low, Close, Prev Close and Volume and many other variables that are not needed. The Close of each stock can be the target, and the Open, High, Low, Prev Close, Volume can be inputs. In order to not include structure break in the data set, we should pick a relatively stable time period, which can be approximately two months.

Solution Statement

In this project, I would like to use recurrent neural networks to solve the problem. I will use **LSTMs (Long-short term memory network as the model)**, the stock that the user chooses as the target, and all the historical time series data of the target stock itself and other stocks as inputs.

I will use the sequence length of about 2 months as the length of the LSTM network (this will be a hyperparameter). The prediction will be the one-step-ahead stock price of the target stock. Once the model is trained, the user can choose what is the 1-month or 2-month period they want to take as input, and the model can predict the next date's adjusted close price.

Benchmark Model

The Benchmark model for this project could be the Linear Regression Model. Linear Regression is a machine learning algorithm based on **supervised learning**. It performs a regression task.

Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting.

This benchmark will use exactly the same input as our LSTM network model and provide benchmark performance for the LSTM.

Evaluation Metrics

This is a classic regression problem therefore best would be to use Root Mean Square Error and R2 score as my evaluation metrics. Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors).

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum (\hat{Y}_i - Y_i)^2}$$

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.

And, R-square can provide how much variation in the dependent variable can be explained by the variation in the independent variables


$$R^2 = \frac{\text{Variance explained by the model}}{\text{Total variance}}$$

R-squared measures the strength of the relationship between your model and the dependent variable on a convenient 0 – 100% scale.

Project Design

The project emphasizes on using only the most correlated features in determining our output. The **nifty50 stock market data** has stock records of 50 companies from year 2000-2019 and has 12 features out of which we will use only 4 features namely, **open** which is the first price at which the security first updates upon the opening of an exchange, **high** which is the highest price at which a stock traded during the course of the trading day, **low** which is a security's intraday low trading price, **volume** which is the amount traded on a particular day.

A model will be built separately for each of the stock companies. Pandas dataframe object would be used to store the dataset and calculations would be done using the numpy library. The data from the year 2000-2016 would be used as **training data** and data from



the year 2017-2019 would be used as **test data**. We would use an LSTM neural network with approximately 180 nodes for input, 3 hidden layers of dimensions 200x150x200 with 200 neurons each and a single node for output. The activation function being used is RELU and I will use RMSE(root mean squared error) and r^2 for checking the accuracy. The scikit linear regression library would be used for our benchmark model. Once the model is built, the user would select a 45-day range from the test data as input and then the close value of the very next day would be our expected output.

Reference

- ❑ Murtaza Roondiwala, Harshal Patel and Shraddha Varma “Predicting Stock Prices Using LSTM” (2015): 78.96
- ❑ Hochreiter, Sepp, and Jürgen Schmidhuber— Long short-term memory, || Neural computation 9.8 (1997): 1735-1780