

Machine Learning Engineer Nanodegree

Capstone Proposal

by Wonho

September 22, 2018

Domain:

Investment and Trading - Build a Stock Price Indicator

Domain Background

The stock price market is barely predictable even though many smart people trying to predict the price. However, there have been many tries to predict the stock price or computational trading from using a statistical method, Support Vector Machines and even deep learning technique these days. Many investment companies are trying to apply machine learning for automated trading systems, and about 70% of all orders on Wall Street are placed by software. As the machine learning technology is improved, many machine learning engineers are trying to predict stock price with machine learning. There are so many data features which the machine learning to use of, and it's hard to choose appropriate features. In this capstone project, features related stock price would be used to predict the stock price of some corporations.

Problem Statement

The goal of this project is to predict the stock price by using time series data related to the specific corporation. However, since it is almost impossible to predict specific price in specific future time, the predicted price is around the real price with some error. So I'm going to predict the future price of a selected company with some error by training with historical stock price data given by input value.

There are two interfaces for the stock predictor. One is a training interface. The training interface accepts a data range and a list of ticker symbol for training machine learning model. Moreover, the second interface is a query interface which accepts a list of dates and a list of ticker symbols and outputs the predicted stock prices for each of those stocks on given dates.

Datasets and Inputs

For this project, I will use the public traded company data from IEX API. Moreover, the companies would be in the S&P 500. By using Python module pandas-datareader, it is possible to query what I need. I need data contains Open, Close, High, Low, and Volume for the companies and it seems that IEX API provides a stable API for developers about the needed data.

Solution Statement

In this project, I will build a price predictor for S&P 500 companies which is trained by prices of some period and takes the target date as input. Moreover, the predictor predicts the

closing price of the input DateTime with the recurrent neural network. I use LSTM for RNN model. The LSTM stands for Long Short-Term Memory model which capable of learning from time series data. The target date or date to predict should be the date after the training date period. I also use the daily training data from the training interface, and the prediction date or target date would be a future date after now and displaying on the query interface of the web application. The price we will predict is the closing price of the company in S&P 500 on the target date.

Benchmark Model

I will use the linear Regression model for benchmark model which is a straightforward supervised learning model. The regression model will be built in sci-kit learn, and the other condition is same as LSTM model which is a solution model for this project. The prediction of the benchmark model is also measured by evaluation metrics below.

Evaluation Metrics

One of the evaluation metrics is root-mean-square-error because we should compare the companies in S&P 500 price between the actual and predict values. Root Mean Square Error or RMSE is considered to be one of the most popular metrics for the regression problem.

Project Design

0. Setup Infrastructure

using iPython Notebook

install requiring Libraries such as Keras, Pandas, Matplotlib, Sklearn, Numpy.

Git Project organization

1. Prepare Dataset

incorporate data of S&P 500 companies with IEX API

Process the requested data into Pandas Dataframe

Develop function for normalizing Data(price and volume)

Dataset will be used with an 80/20 split on training and testing data

2. Develop Benchmark Model

Set up basic Linear Regression model with Scikit-Learn

Calibrate parameters

3. Develop Basic LSTM Model

Set up basic LSTM model with Keras utilizing parameters from Benchmark Model

4. Improve LSTM Model

Develop, document, and compare results using additional labels for the LSTM model

5. Document and Visualize Results

Plot Actual, Benchmark Predicted Values, and LSTM Predicted Values per time series

Analyze and describe results for a report

6. make a web application with Node.js, MongoDB.

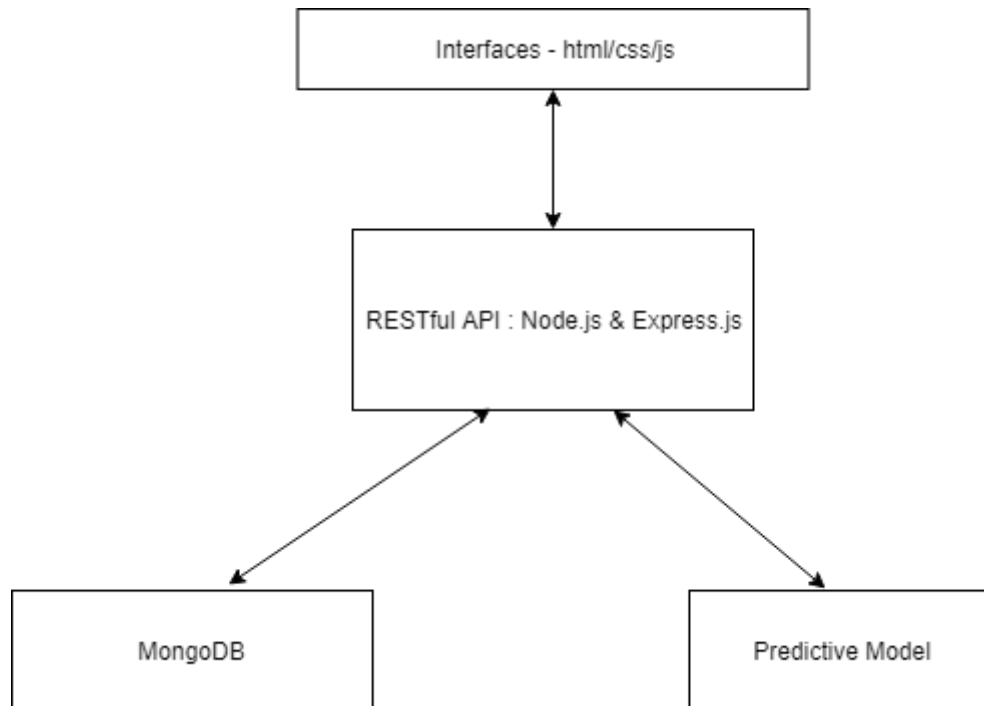
Environment setup

Node.js RESTful API implementation

Create DB for tracking previous predictions and results

Create two interfaces which one for training and the other for query interface.

Here is the structure of a stock predictor web application.



References

Root Mean Square Error(wiki) - https://en.wikipedia.org/wiki/Root-mean-square_deviation

IEX API - <https://iextrading.com/developer/>

LSTM(wiki) - https://en.wikipedia.org/wiki/Long_short-term_memory

Stock Price predict Youtube Video:

Stock price prediction | AI in Finance -

<https://www.youtube.com/watch?v=7vunJlqLZok&t=160s>

How to Predict Stock Price easily - <https://www.youtube.com/watch?v=ftMq5ps503w&t=16s>