Li, DING

16 November 2017

Machine Learning Capstone Project Proposal

A Stock Price Indicator

Domain Background

I will build a stock price indicator with data from yahoo. Big finance companies have used computer algorithm to trade automatically long time ago, and especially on strategy. They often use linear regression, or more complex game-theoretic and pattern recognition, or even Markov Chain Monte Carlo[1]. With the emerge of deep learning, this should be a solid way to solve trade strategy more efficiently. Many researchers have discussed on this topic, such as University Berkeley's Predicting Stock Market Movement with Deep RNNs [2] . In this project, I will use the RNN to predict stockers on specific data mentioned above.

Problem Statement

The essence of stock trading is sell at high price and buy at low price. So here, we define the problem as Predict the stock price given the history prices of this stock, use modern technology machine learning or deep learning.

Table 1: Data Sample

Date	Open	High	Low	Close	Adj Close	Volume
2017-10-16	25.299999	25.51	25.25	25.43	25.353802	56900
2017-10-17	25.48	25.48	24.99	25	24.925091	97900
2017-10-18	25.07	25.23	24.969999	24.99	24.915119	36500

[1] https://en.wikipedia.org/wiki/Algorithmic_trading#Strategy_implementation

[2] https://bcourses.berkeley.edu/files/70257274/download/

Datasets and Inputs

The input data are some companies (such as IBM, APPLE or GOOGLE) stock data range from 2011 to 2015 from yahoo finance. Table1 is a sample data from yahoo finance. and I will use 'Date', 'High', 'Low', 'Close' and 'Volume' for prediction. The meaning of those columns are described on Table2.

Table 2: Data Description

Column Name	Data Value	Description
Date	Date	The stock trade date
High	Float	The stock highest prices on the given date
Low	Float	The stock lowest prices on the given date
Volume	Int	The trade stock number on the given date

Solution Statement

I will use neural network for on this project, first split roughly 80% to 20% on train-test split specified by the dataset. In order to preserve the sequential nature of the data, the split is not random; rather, the first 2 years of data are used as the training set and the last 1 year is reserved for the test set. And then the data will be cleaned, such as normalized, before feed into neural network. After that, I will use LSTM or GRU as hidden layer, and dropout in case of over-fitting. The loss function is cross entropy or MES to best fit the variance of model. And learning rate, range from 0.01 to 0.001, is upon real experiment.

Benchmark Model

Here I define average price = (high price + low price) /2.

I will use last seven days moving average[3] as benchmark to prove the RNN model's effectiveness or not. The formula is as below, where X1 - X7 are last seven days average price:

$$today\_price = \frac{X_1,\ldots,X_7}{7}$$

[3] https://en.wikipedia.org/wiki/Moving_average

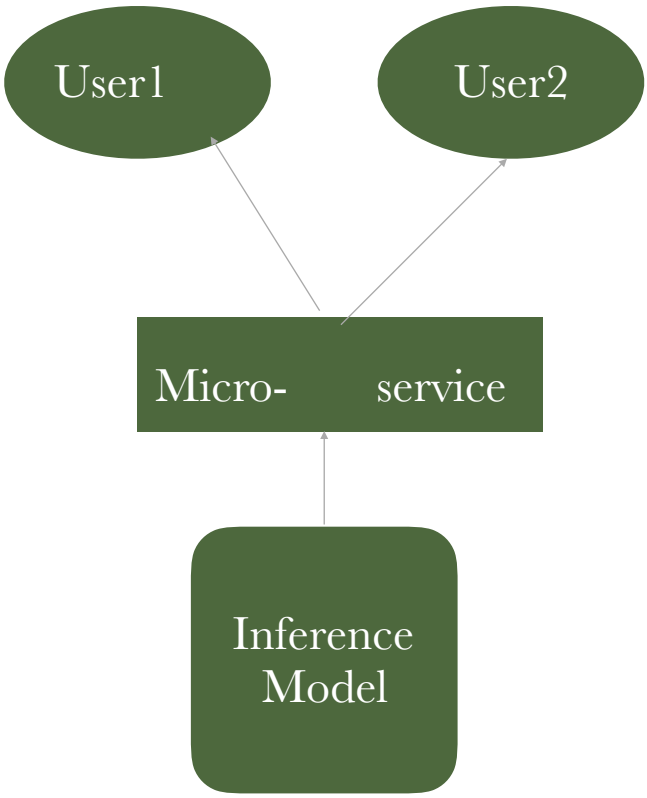
Evaluation Metrics

The final model will be evaluated by the average stock prices on RMSE[4] , where Y is the predict value and y is the real average price.

$$RMSE = \frac{1}{n}\sqrt{\sum (Y - y)_i^2}$$

Project Design

The whole project is designed as micro service architect. The user can use service to predict one week stock prices given previous week's stock price. At first, user select which company they are will to predict (that is IBM, APPLE or GOOGLE), and then input one weeks record including date, high, low price and volume, then the service return the predicted next weeks average stock price, day by day. Below briefly illustrate the structure on this workflow.



[4] https://en.wikipedia.org/wiki/Root-mean-square_deviation