New York Stock Exchange Predictions

This repository is about predicting the stock prices of a single stock type (in this case Apple with symbol 'APPLE') by using the Long-Short Term Memory (LSTM) networks .The kaggle data is used for this study available from 2012 to 2016 for S & P 500 companies.

Data loading and preprocessing:

The data file ' prices-split-adjusted.csv' is loaded as a data frame which has the columns 'open', 'close','high', 'low', 'volume' for a given company ticker 'symbol'. The 'date' column is set as the index column when loading the data.

date	symbol	open	close	low	high	volume
2016-01-05	WLTW	123.43	125.839996	122.309998	126.25	2163600.0
2016-01-06	WLTW	125.239998	119.980003	119.940002	125.540001	2386400.0
2016-01-07	WLTW	116.379997	114.949997	114.93	119.739998	2489500.0
2016-01-08	WLTW	115.480003	116.620003	113.5	117.440002	2006300.0
2016-01-11	WLTW	117.010002	114.970001	114.089996	117.330002	1408600.0

Data manipulation:

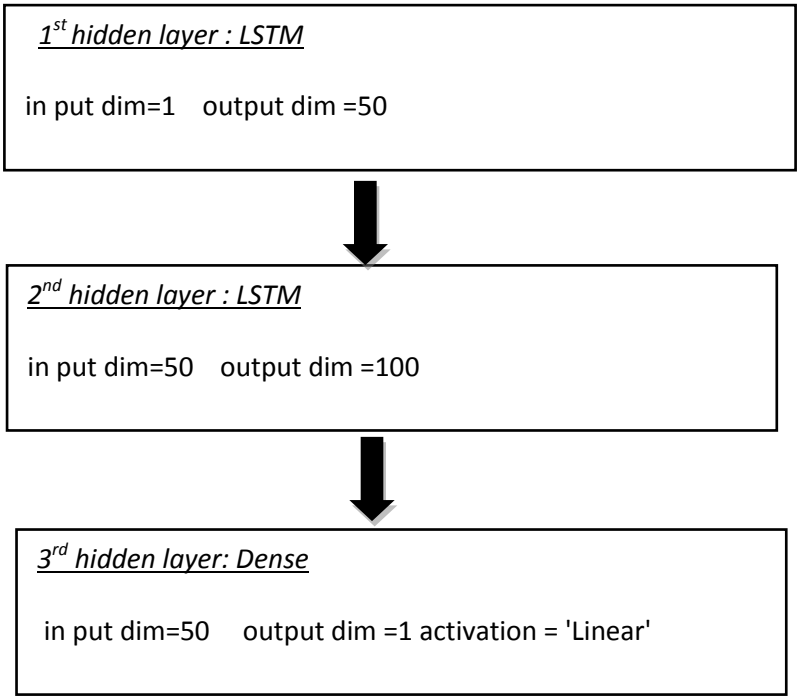
The unique symbol names are extracted and a separate data frame is created for symbol 'AAPL' with 'close' column. The purpose is to predict the closing price. 50 new columns are created and next 50 data points are entered one by one for each date(index).This creates a 51 day sequence .Then these values are normalized for better convergence and to be compatible with the activation functions in the neural network. The first 50 day sequence acts as a input sample (X) while 51st day acts as the label (y).

	AAPL_close	1	2	3	4
date					
2010-01-04	30.572857	30.625713	30.138571	30.082857	30.282858
2010-01-05	30.625713	30.138571	30.082857	30.282858	30.015715
2010-01-06	30.138571	30.082857	30.282858	30.015715	29.674286
2010-01-07	30.082857	30.282858	30.015715	29.674286	30.092857
2010-01-08	30.282858	30.015715	29.674286	30.092857	29.918571
2010-01-11	30.015715	29.674286	30.092857	29.918571	29.418571
2010-01-12	29.674286	30.092857	29.918571	29.418571	30.719999
2010-01-13	30.092857	29.918571	29.418571	30.719999	30.247143
2010-01-14	29.918571	29.418571	30.719999	30.247143	29.724285
2010-01-15	29.418571	30.719999	30.247143	29.724285	28.250000
2010-01-19	30.719999	30.247143	29.724285	28.250000	29.010000
2010-01-20	30.247143	29.724285	28.250000	29.010000	29.420000
2010-01-21	29.724285	28.250000	29.010000	29.420000	29.697144
2010-01-22	28.250000	29.010000	29.420000	29.697144	28.469999
2010-01-25	29.010000	29.420000	29.697144	28.469999	27.437143

This table runs up to 50 columns

Training the neural network:

To train the model RNN are used due to the reason that the stock data has a time component and it can be framed as a time series supervised learning problem. The x_t sample value depends on the x_{t-1} and previous data points and LSTM model is used to create the network. Keras module (Sequential model) is used for this study which runs on Tensor Flow backend. The schematic diagram which demonstrates the used neural network model is shown below.



The **input data (X)** are in shape: [Nu of samples, sequence length, nu of features] which in this case is [Nu of samples, 50,1].Following the same pattern the **label data (y)** are in shape [Nu of samples, 1 ,1]. When training the batch_size is set to 1 for smoother training. As loss function "mean square error' is used.

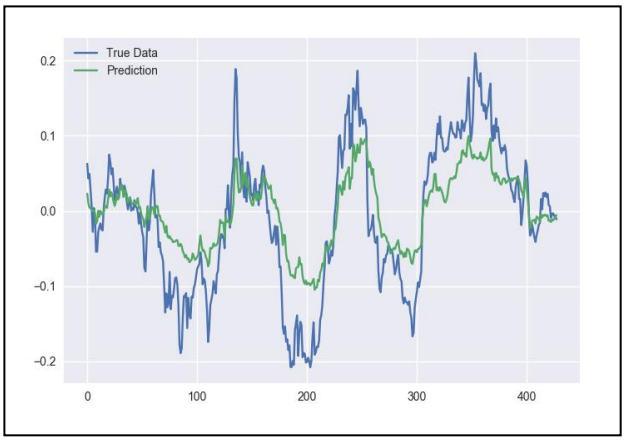
Predictions:

The prediction shows the score of : 0.0007 with the following figure. Batch size= 1 and epochs=10 are used to generate this prediction.

epochs =10	batch size= 1
Score: 0.000327119	

Hyper parameter tuning.

Prior to achieve the above prediction , the batch size was changed from 512 to 1 while keeping the epochs=1 fixed.



epochs =1	batch size= 512
Score: 0.003106	

epochs =1	batch size= 1
Score: 0.00138711	