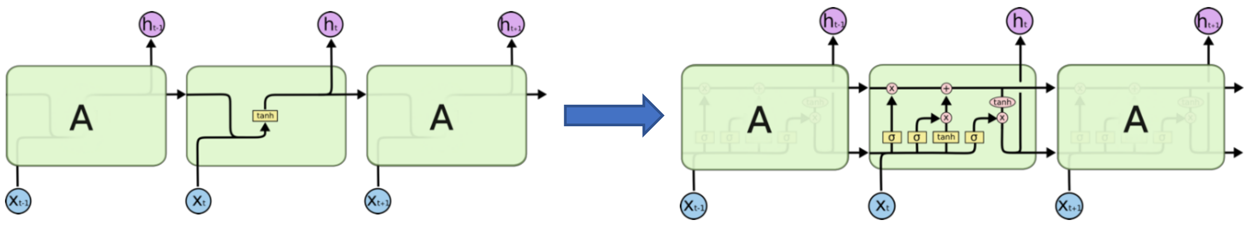
Stock Price Trend Prediction Using Machine Learning Through Python

1. Introduction

This system contains the following four parts: data processing, LSTM model building, training and testing, and data visualization.
TensorFlow is machine learning python package used.
LSTM is another vital tool. LSTM means long-short term memory. It’ s one of RNN models.
The key to LSTMs is the cell state, the horizontal line running through the top of the diagram. The cell state is kind of like a conveyor belt. It runs straight down the entire chain, with only some minor linear interactions. It’ s very easy for information to just flow along it unchanged. The LSTM does have the ability to remove or add information to the cell state, carefully regulated by structures called gates. Gates are a way to optionally let information through. They are composed out of a sigmoid neural net layer and a pointwise multiplication operation.



Normal RNN cell LSTM cell

2. Python Implement

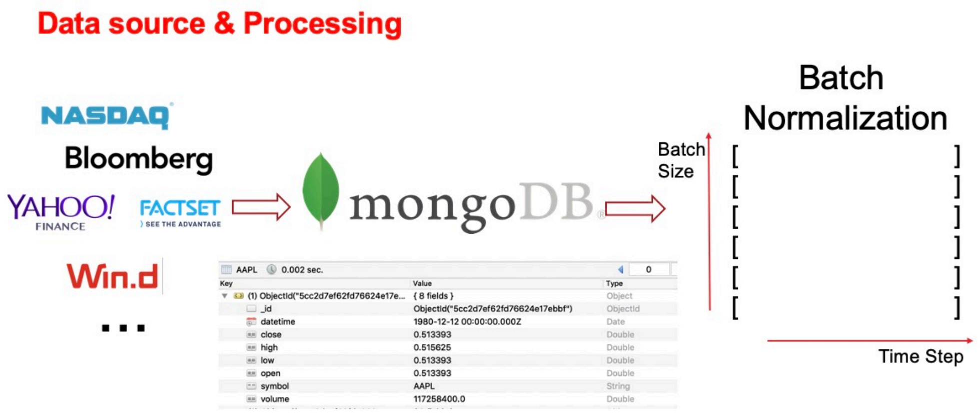
Four main packages: **TensorFlow, NumPy, Pandas, and plotly**.

2.1. Data processing

Considering that there are many kinds of data sources which have different formats, so I developed a data process class to deal with all data sources such as Bloomberg, yahoo finance and so on. Then stored data in a standard format into MongoDB. The reason to do that because in this way, we don't need to modify our code according different data sources, which's more convenient.

Then a Data_Generator class is developed to make data got by database into data needed by model learning. The core function is get_batch(steps), we can get this kind of data. the data shape is batch_size* time_steps .

Tensorflow's lstm cell can read data by one batch instead of one sequence, it's more efficient. At time_step 1, model read 1 and to predict 2, at time_step 2, model read 2 and to predict 3 and meanwhile it keeps the effect from time_step1, this's is how long and short memory works.



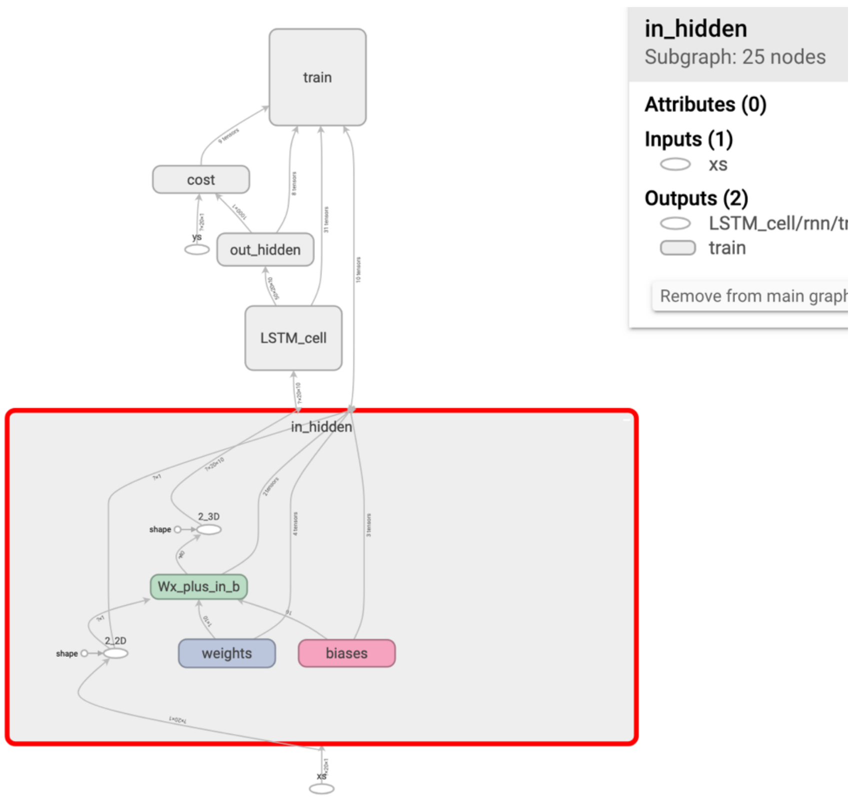
AAPL 0.002 sec. 0

Key	Value	Type
(1) ObjectId("5cc2d7ef62fd76624e17e...	{ 8 fields }	Object
_id	ObjectId("5cc2d7ef62fd76624e17ebbf")	ObjectId
datetime	1980-12-12 00:00:00.000Z	Date
close	0.513393	Double
high	0.515625	Double
low	0.513393	Double
open	0.513393	Double
symbol	AAPL	String
volume	117258400.0	Double

2.2. LSTM model building

At the beginning is data has been dealt with Data_generator, its shape is (batch_size, time_step,1), one means one factor. Here our factor is the close price of stock. Then our data will go through input hidden layer, then it enters the LSTM_cell. In the end, it will get the result going out of output hidden layer.

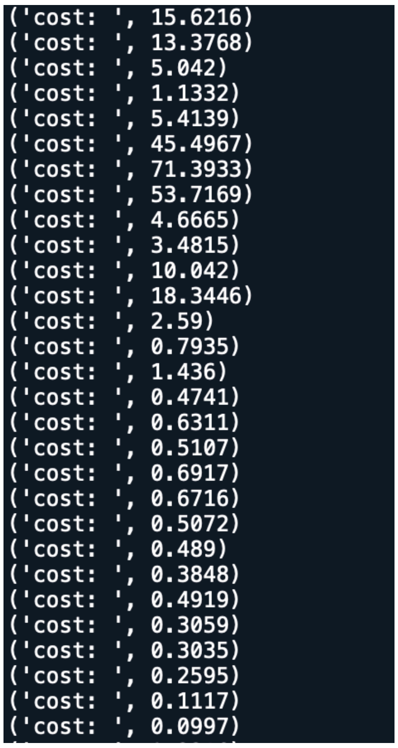
Cell_size is the numbers of cells in each lstm cell. LR is the learning rate which work for model optimizer, we can choose it, but not too large, by several modification. 0.006 can get pretty good result. Split is the coefficient which



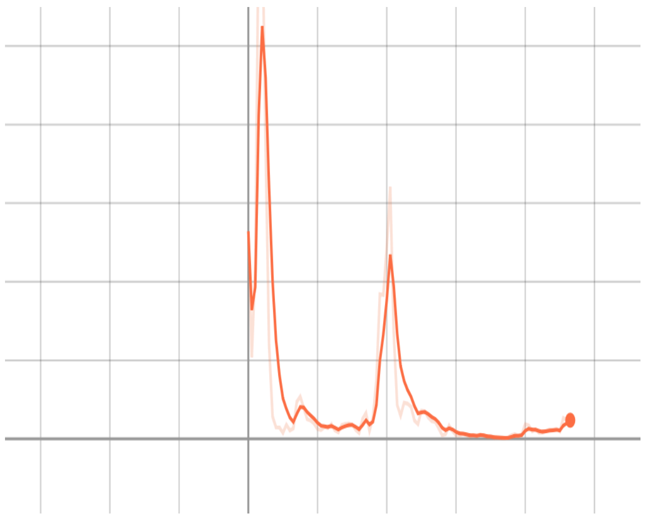
cut data into train set and test set. Here we 80 percent of data is train set and 20 percent of data is test set. DB_NAME is our database name.

Here is machining learning model designed, which includes three parts: input hidden layer, lstm layer and output hidden layer. In two hidden layers, using 10 random weights and biases to process data and to feed and receive data into or from lstm layer. From experiments, we can see this model can get better results than only single lstm layer.

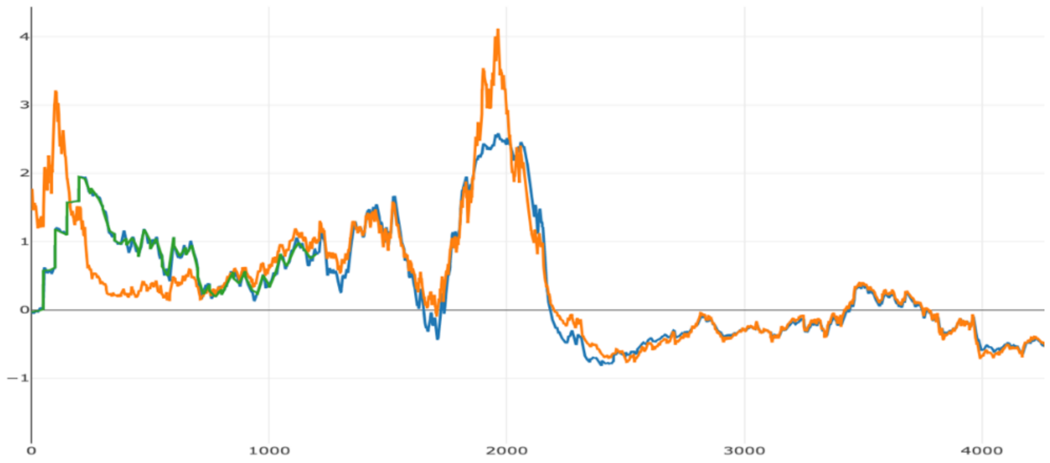
After we build this model, we need to try to reduce loss of real data. Making the loss function: mean-square error, it can calculate the error between real data and prediction data. Then to use the AdamOptimizer() to optimize the coefficient of model. Then we can find in each iteration, the error decline, it means that the model is learning something



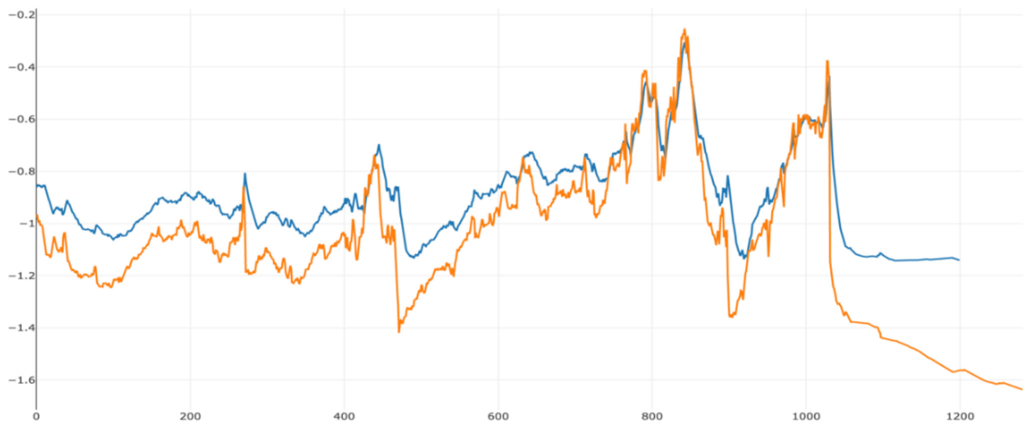
```
('cost: ', 15.6216)
('cost: ', 13.3768)
('cost: ', 5.042)
('cost: ', 1.1332)
('cost: ', 5.4139)
('cost: ', 45.4967)
('cost: ', 71.3933)
('cost: ', 53.7169)
('cost: ', 4.6665)
('cost: ', 3.4815)
('cost: ', 10.042)
('cost: ', 18.3446)
('cost: ', 2.59)
('cost: ', 0.7935)
('cost: ', 1.436)
('cost: ', 0.4741)
('cost: ', 0.6311)
('cost: ', 0.5107)
('cost: ', 0.6917)
('cost: ', 0.6716)
('cost: ', 0.5072)
('cost: ', 0.489)
('cost: ', 0.3848)
('cost: ', 0.4919)
('cost: ', 0.3059)
('cost: ', 0.3035)
('cost: ', 0.2595)
('cost: ', 0.1117)
('cost: ', 0.0997)
```



2.3. training and testing



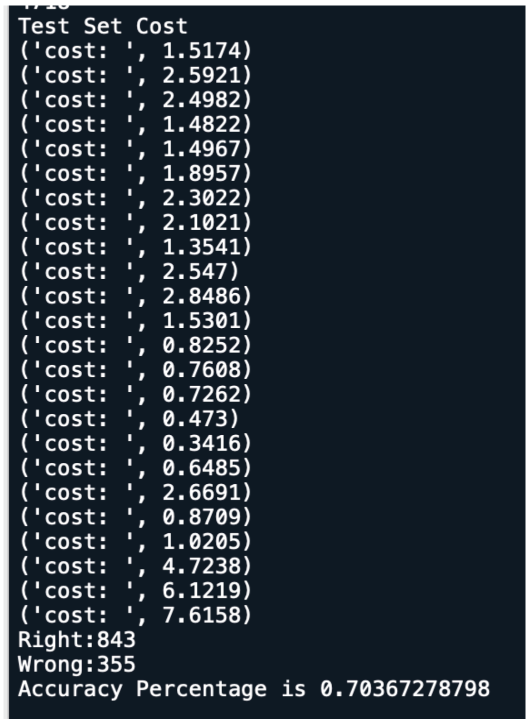
This is training set result. At the beginning, we can see our prediction curve cannot fit the real data well. But after many times training, we can see the prediction show good result.



This is testing set result. Also, the prediction can fit real data curve very well.

3. Output Analysis

According the result predicted by the model using the testing set data, we can see the cost(error) between real data and prediction data keep as small as what when we used training set data which is around 0-2, which explains what we can see from the graph: the prediction curve can fit the curve pretty good. The point is the trend of two curves are the same, which is we are going to make use. We calculate the accuracy rate of prediction. If the price of real data and prediction data go up or down both, we make it right. If the price go to different directions, we make it wrong, then to summarize the result to output. As we can see, the accuracy is around 70%, when making trading strategy, we can use this model as our assistant tool.



```
Test Set Cost
('cost: ', 1.5174)
('cost: ', 2.5921)
('cost: ', 2.4982)
('cost: ', 1.4822)
('cost: ', 1.4967)
('cost: ', 1.8957)
('cost: ', 2.3022)
('cost: ', 2.1021)
('cost: ', 1.3541)
('cost: ', 2.547)
('cost: ', 2.8486)
('cost: ', 1.5301)
('cost: ', 0.8252)
('cost: ', 0.7608)
('cost: ', 0.7262)
('cost: ', 0.473)
('cost: ', 0.3416)
('cost: ', 0.6485)
('cost: ', 2.6691)
('cost: ', 0.8709)
('cost: ', 1.0205)
('cost: ', 4.7238)
('cost: ', 6.1219)
('cost: ', 7.6158)
Right:843
Wrong:355
Accuracy Percentage is 0.70367278798
```

4. **Discussion**

I believe that this forecast results are far more than practice applying. It’ s impossible for us to make long short decision from the machine prediction. But we have reasons to believe that the forecast results can be used to assist in decision making.