Using LSTM(s) to Accurately Predict Stock Market Prices

Ruben Malvaez, Muhammad Rehan, Ghada O Al Kaabi

Department of Computer Science, University of Houston

Abstract

As financial institutions start to embrace artificial intelligence, the use of machine learning is increasingly utilized to help make trading decisions. One of the most intelligent data mining methods, known as the neural network, has been used by many data scientists and investment analysts to observe the behavior of stocks in the future. The primary objective of this project is to expose our audience with neural network's application in the financial industry, which can be beneficial for their future investment decisions in stocks. In this paper, we applied an improved version of recurrent neural networks (RNN) called as long short-term memory neural network (LSTM) for times series forecasting. It is better suited for our aim to predict the stock prices, as LSTMs tends to retain the information over a prolonged period. The four data sets we used are extracted from New York Stock Exchange through yahoo finance API, roughly dated from 2010 to 2016. Moreover, we used Python programming language libraries (Keras, TensorFlow) to implement the neural network, along with the other libraries (NumPy, Pandas, Matplotlib, Seaborn) to visualize, manipulate and preprocess the data.

Background

- The motivation is to predict stocks with the highest accuracy using the LSTM neural network. The main goal would be to learn and ensure the highest chance of automated profitability possible and make money.

- The reason for choosing an LSTM network is basically to enable you to model time-dependent and sequential data problems. Therefore, we chose to demonstrate the future price prediction for different stocks using recurrent neural networks (RNN) in TensorFlow.

- The RNN will take the input from two sources, one is from the present and the other from the past. Information from these two sources are used to decide how they react to the new set of data. This is done with the help of a feedback array where output at each instant is an input to the next moment. so that the recurrent neural network has memory.

Methods

We chose an LSTM neural network due its ability to process entire sequences of data instead of a single datapoint. This feature is ideal since we are analyzing streams of stock's closing prices over long periods. LSTM is also better than other RNN networks as it does not suffer from information loss (gradient vanishing) within its neuron connections as the amount of recursion increases.

- **5-layers** deep network plus output layer
- Two **30% dropout layers** that help us avoid overfitting by randomly dropping out 30% of the neurons in the layer
- A final layer that returns the result for the predicted price of the next day
- We decided to run the network at **80 epochs** after thorough testing with the goal to minimize error and maximize efficiency
- Used **MSE (mean squared error)** to maximize the correction with big errors between the predicted values and the training set. This allows us to increase the accuracy of the network very quickly is a few epochs as opposed to using something like **mean absolute error** which does not punish big errors more heavily.

We compared our predicted closing price results to the real-world stock exchange data. We measured which stock factors were essential to develop superior accuracy and we landed on 3 parameters:

Volatility – The rate at which the price of a stock fluctuates (increases or decreases) around its average price over a particular period.

Stock Trend – The overall movement of the stock price, its either increasing (bullish) or decreasing(bearish)

Training Data Time Frame – We used data points dating back 3 and 5 years back to train our model and improve its future predictions accuracy.

Results

- Volatility Comparison for Predicted vs. Real World

	S&P 500 (Low Volatility)	Vizsla Silver Corp (High Volatility)
Volatility	≈ 1%	≈ 9%
5 Day Percent Change Between **Real** Peak and Low Prices	2.85406%	8.3871%
5 Day Percent Change Between **Predicted** Peak and Low Prices	1.27817%	8.94309%

The volatility of the model's predictions closely resembled the historic volatility of the stock with which the model was trained. This implies that there is a conservation of volatility by the neural network which is great for low volatility stocks since they are usually not greatly affected by minor outside factors (i.e.. S&P 500). On the other hand, the network is naive when presented with external factors that may suddenly affect a stock price as it does with penny stocks such as Vizsla Silver Corp.

- Comparison between Microsoft and Apple Stock

The model performed quite well when the graph for the stocks showed a bearish trend. However, the model's performance tends to get weaker when we see a bullish trend for the Apple and Microsoft stocks.

Conclusion

- Generally, our predictions seem to be the most accurate and reliable when we use the network with a conjunction of **low volatility** stocks in a **bearish trend** and using a **five-year data period** training set.
- Using the three optimization tactics for choosing potential stocks for our model will ensure much higher accuracy and in turn better possibilities for profitability while minimizing user analysis and input.

Future Direction

- Tweak the LSTM model to increase the overall accuracy of it by lowering down the mean squared error.
- Optimize the neural network to perform better during bullish and/or bearish trends of the stock market prices.
- Develop and compare the LSTM's performance to predict the stock market prices with other neural networks such as CNN and MLP.

Acknowledgments

- Teo, Bee Guan. "Stock Prices Prediction Using Long Short-Term Memory (LSTM) Model in Python." Medium. The Handbook of Coding in Finance, October 26, 2021. https://medium.com/the-handbook-of-coding-in-finance/stock-prices-prediction-using-long-short-term-memory-lstm-model-in-python-734dd1ed6827