Prakhar Tripathi
@prakhar.tripathi2205@gmail.com

Udacity Machine Learning Nanodegree(MLND)

Proposal for Predicting Stock Market Prices of AMD using LSTM

February 24,2019

(NASDAQ)

Domain Background

Predicting the Stock Market has been the bane and goal of investors since its existence. Everyday billions of dollars are traded on the exchange, and behind each dollar is an investor hoping to profit in one way or another. Entire companies rise and fall daily based on the behaviour of the market. Should an investor be able to accurately predict market movements, it offers a tantalizing promises of wealth and influence. It is no wonder then that the Stock Market and its associated challenges find their way into the public imagination every time it misbehaves. The 2008 financial crisis was no different, as evidenced by the flood of films and documentaries based on the crash. If there was a common theme among those productions, it was that few people knew how the market worked or reacted. Perhaps a better understanding of stock market prediction might help in the case of similar events in the future.

Can we actually predict stock prices with machine learning? Investors make educated guesses by analyzing data. They'll read the news, study the company history, industry trends and other lots of data points that go into making a prediction. The prevailing theories is that stock prices are totally random and unpredictable but that raises the question why top firms like Morgan Stanley and Citigroup hire quantitative analysts to build predictive models. We have this idea of a trading

2.

floor being filled with adrenaline infuse men with loose ties running around yelling something into a phone but these days they're more likely to see rows of machine learning experts quietly sitting in front of computer screens. In fact about 70% of all orders on Wall Street are now placed by software, we're now living in the age of the algorithm.

This project seeks to utilize Deep Learning models, Long-Short Term Memory (LSTM) Neural Network algorithm, to predict stock prices. For data with timeframes recurrent neural networks (RNNs) come in handy but recent researches have shown that LSTM, networks are the most popular and useful variants of RNNs.

I will use Keras to build a LSTM to predict stock prices using historical closing price and trading volume and visualize both the predicted price values over time and the optimal parameters for the model.

Problem Statement

The challenge of this project is to accurately predict the future closing value of a given stock across a given period of time in the future. In the past few years we've seen lots of academic papers published using neural nets to predict stock prices with varying degrees of success but until recently the ability to build these models has been restricted to academics.During the training of RNN, as the information goes in loop again and again which results in very large updates to neural network model weights. This is due to the accumulation of error gradients during an update and hence, results in an unstable network.. Now with libraries like tensor flow anyone can build powerful predictive models trained on masive of datasets. For this project I will use a Long Short Term Memory networks - usually just called "LSTMs" to predict the closing price of the S&P 500 using a dataset of past prices

Goals

1. Explore stock prices.
2. Implement basic model using linear regression
3. Implement LSTM using keras library.
4. Compare the results and submit the report

Datasets and Inputs

I'll be using the daily prices of the S&P 500 from May 2009 to August 2018, this is a series of data points indexed in time order or a time series. My goal will be to predict the closing price for any given date after training. All of the necessary data for the project will come from Google Finance.

Solution Statement

For this project according to my research the best possible solution is to utilize a LSTM Neural Net model capable of learning from time series data. This project will be programmed in a Jupyter Notebook (iPython) for ease of reproducibility. Using a Keras implementation of the Tensor Flow library, the solution will utilize a LSTM Neural Net model and will be supported by Pandas DataFrame library for convenient time series data schema.The measures of performance will be based on the predicted stock ticker price in comparison to both the actual price and the benchmark model's predicted price.

Benchmark Model

For this project i will use a Linear Regression model as its primary benchmark. As one of my goals is to understand the relative performance and implementation

differences of machine learning versus deep learning models. This Linear Regressor will be based on the examples presented in Udacity Machine Learning Nanodegree(Boston house prediction) and will be used for error rate comparison MSE and RMSE utilizing the same dataset as the deep learning models.

Evaluation Metrics

For this project i will measure performance using the mean squared difference between predicted and actual values of the target stock at adjusted close price and the delta between the

performance of the benchmark model (Linear Regression) and our primary model (Deep Learning).

Project Design

This project will be implemented through the Keras/Tensor Flow library using LSTM Neural Networks. Development workflow will follow the below sequence:

0. Set Up Infrastructure

- iPython Notebook
- Incorporate required Libraries (Keras, Tensor flow, Pandas, Matplotlib, Sklearn, Numpy)
- Git project organization

1. Prepare Dataset

- Incorporate data of S&P 500 companies
- Process the requested data into Pandas Dataframe
- Develop function for normalizing data
- Dataset will be used with a 80/20 split on training and test data across all models

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 LSMT model

5. Document and Visualize Results

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

- Analyze and describe results for report.

NOTE : The actual implementation may vary a little from proposed design as i have not yet implemented the project. But the flow of working process will be same as mentioned above.

@Prakhar Tripathi