Stock Price Prediction Using LSTM

Team Random

Yaming Cao
Jingzhen Hu
Qingzhong Liang
Arafatur Rahman
A K M Rokonuzzaman Sonet

The Erdos Institute, Summer Boot Camp 2022

Data gathering and Preprocessing

- Historical data for the stocks AAPL, TSLA, AMD, SBUX, FB from Yahoo finance for the period **06/02/2012 – 06/02/2022.**
- Considered only the **daily opening prices**.
- **Testing data consists of last 90 days** (1/25/2022 - 06/02/2022)
- **Remaining s**ample used for the **training** data (06/02/2012 – 1/24/2022)
- **Normalized** training and testing data using sklearn *StandardScaler* package.

Modeling Approach: Long Short Term Memory (LSTM)

Today’s stock price will determined by:

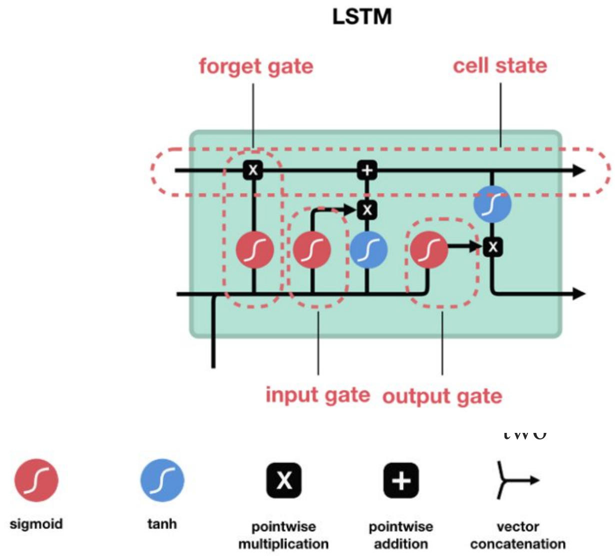
- The **pattern** the stock has been following in the **past days**, which could be **down or up**.
- The **price of the stock on the previous day**, because many traders compare the stock’s previous day’s price before buying it

These relationships can be generalized to any problem as:

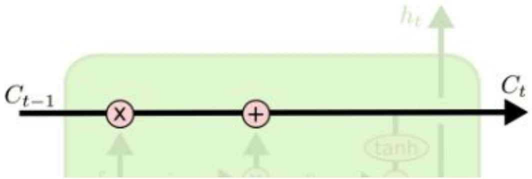
- **The previous cell state:** information present in the memory after the previous time step
- **The previous hidden state:** output of the previous cell state, h_{t-1}
- **The input at the current time step**: new information at that moment, x_t

The Long Short Term Memory model has these features!

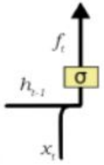
- LSTM is able to **store information** from the past which helps especially predict stock price based on past prices.
- LSTM has **gates capable of regulating** what information to **keep** or **forget.**
- LSTM cell contains a f**orget state, input gate, output gate, and cell state,** along with **activation functions, Sigmoid and tanh.**



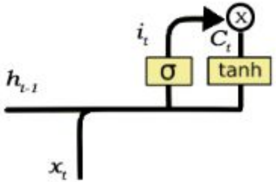
LSTM Structure and Mechanism



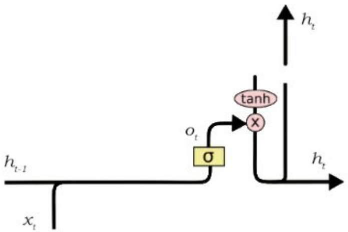
Cell State: working like a memory and key to LSTMs. It carries informations and associated with some linear interactions. LSTM either remove information from this state or add information to this state to carry over.



Forget Gate: decides what information need to forget from the cell state. It takes inputs h_{t-1} and x_t and outputs a number from 0 to 1 for each number in the cell state C_{t-1} . 0 to forget and 1 to input.



Input Gate: add information to the cell state in three steps: (1)Like the forget gate act as a filter for all the information from h_{t-1} and x_t ; (2) Create a vector containing all possible values that can be added to the cell state by using tanh function; (3) multiply the created vector to the value of the regulatory filter and then add this information to the cell state.



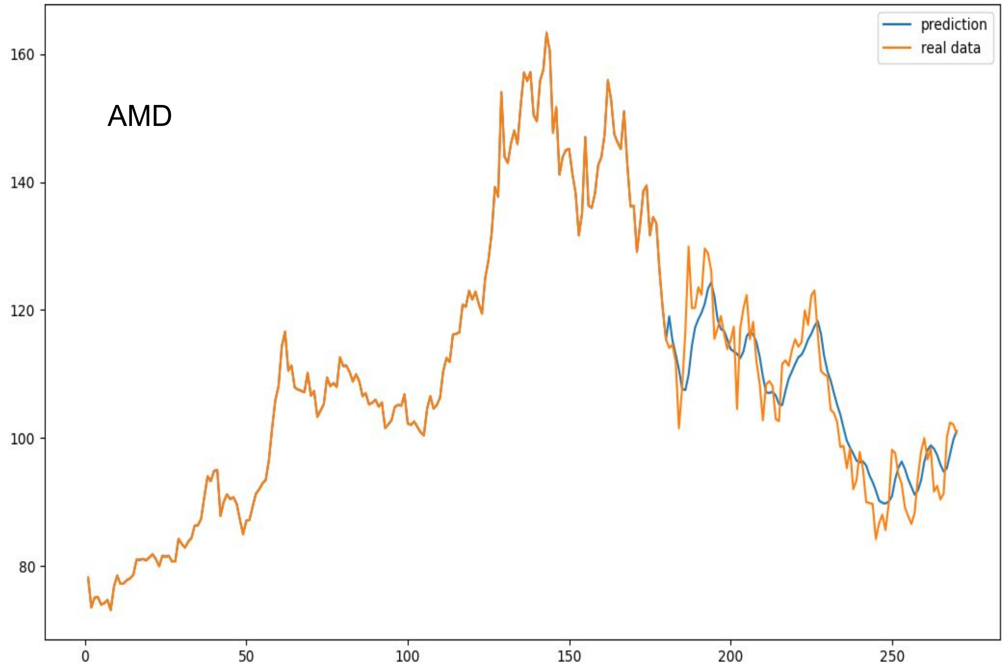
Output Gate: outputs selected useful information from the current cell state in three steps: (1) apply tanh function to the cell state and create a vector; (2) make a filter using the values of h_{t-1} and x_t so that can regulate the values need to be output; (3) multiply the vector created in step 1 to the value of regulatory filter and send it as output and send it to the hidden state of next cell.

Network Architecture

- Model type: **Keras sequential API**
- Two LSTM network with outer spaces **dimension 64 and 32** respectively.
- **Dropout 20%**
- Activation Function: **Relu**; Optimizer: **Adam** ; Loss : **mse**
- **Input space: A 3D tensor** [no of sample, time step, no of features]
- No of **epoch: 3**; batch size: **1**

Empirical Results

Stock Prediction with 14 time steps:



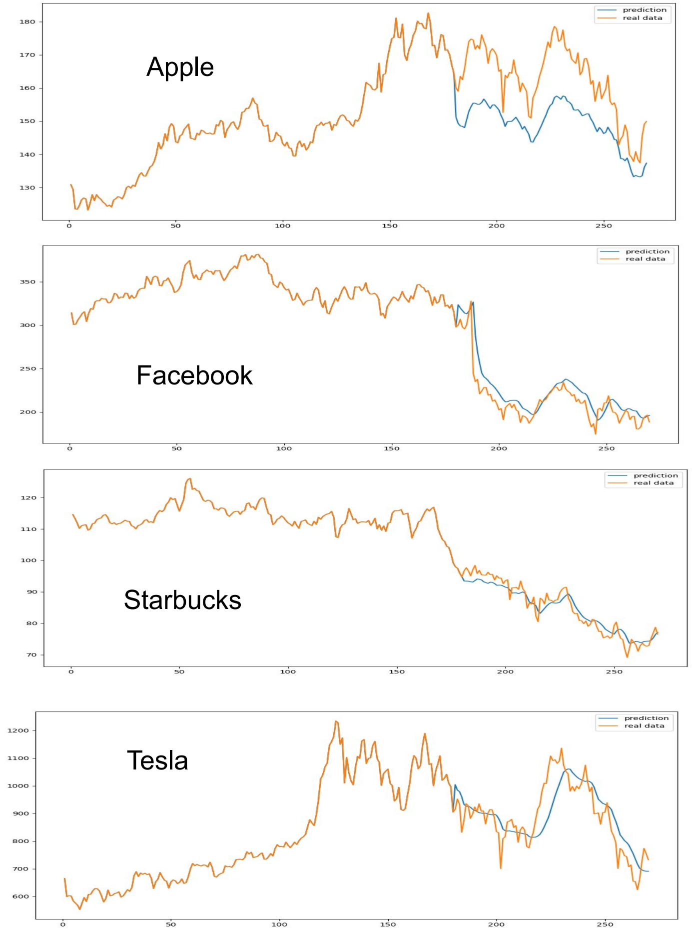
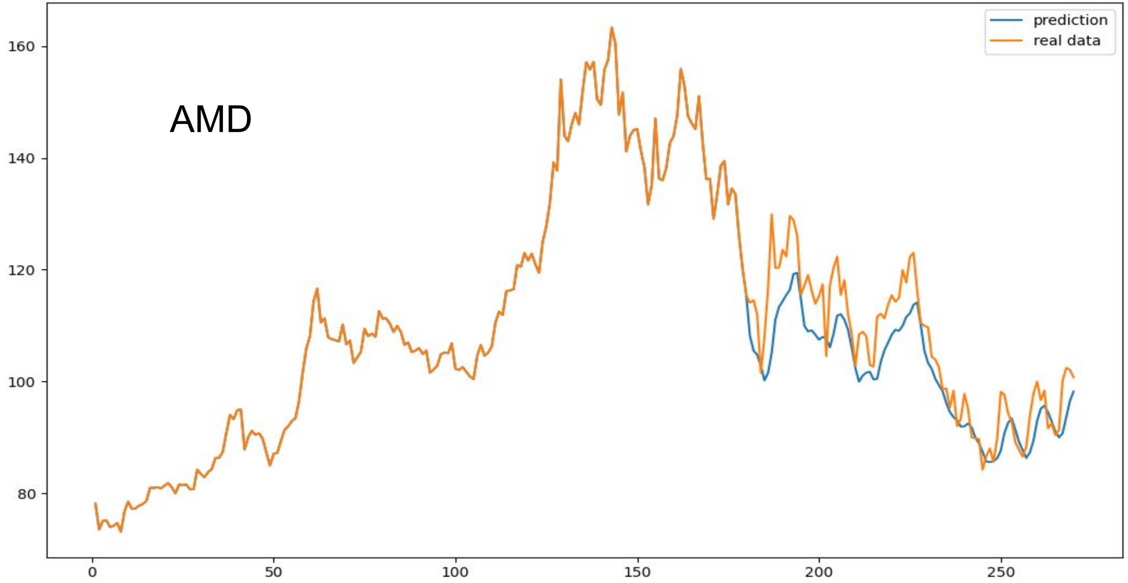
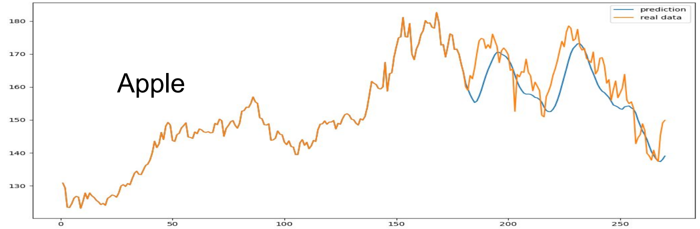


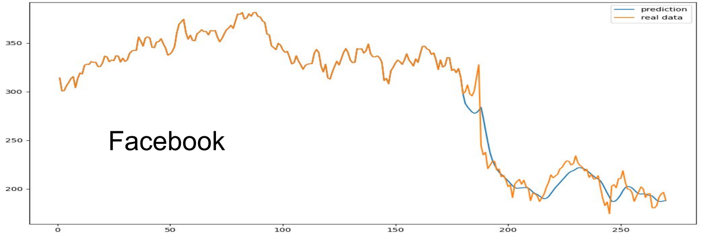
Figure: Actual (Orange) and Predicted (Blue) Stock Price

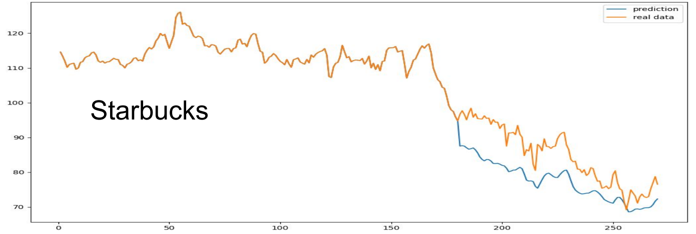
Empirical Results

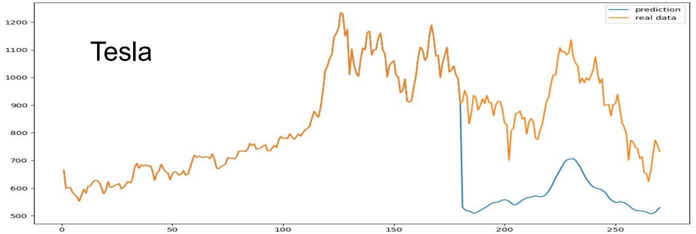
Stock Prediction with 60 time steps:











<u>Figure</u>: Actual (Orange) and Predicted (Blue) Stock Price

Results Discussion

- The predicted value was not very accurate compared to the real price, but it could **capture the direction** of price movement most of the times.
- Among all the stocks, prediction for **AMD** looked more promising and more inaccurate for Tesla.
- **Time steps 14** provided slightly **better** prediction than time **step 60.**

Future Direction

- Can play with **different model architecture,** hyperparameter tuning to see if that increases the performance
- Can do more **exploratory data analysis** to figure out some relation between the model prediction and the properties of data.
- Can do **time series cross validation** to improve the generalization power of our model.
- The predicted results can be used for **portfolio optimization problems**.

References

- **Data: Yahoo Finance**
- Using LSTMs to Predict Future Stock Price
- LSTM Network
- Introduction to LSTM
- Data science approach to stock prices forecasting in Indonesia during Covid-19 using Long Short-Term Memory (LSTM)
- Machine Learning to Predict Stock Price