PREDICTING THE STOCK PRICE OF COMPANIES USING MACHINE LEARNING MODELS

IHEBEDDINE MARNAOUI
COMPUTER SCIENCE ENGINEERING STUDENT
NATIONAL SCHOOL OF COMPUTER SCIENCE, MANOUBA
ihebeddine.marnaoui@ensi-uma.tn

ONES HIZI
COMPUTER SCIENCE ENGINEERING STUDENT
NATIONAL SCHOOL OF COMPUTER SCIENCE, MANOUBA
ons.hizi@ensi-uma.tn

January 8, 2022

ABSTRACT

The stock exchange is a financial market in which stocks and bonds of listed companies are traded. These companies are generally large firms that have real power over the national, and even international, economy. Having a glimpse into the future of the markets and capturing the stock market trend is therefore interesting, even fundamental, for certain trades such as traders who have to make decisions quickly and efficiently. The analysis of stock market activities and more particularly the prediction of stock prices is a problem that has attracted the interest of the scientific community, in which statistics, economics and data scientists have taken an interest. Predicting stock prices is a time series problem since past stock price values are numeric values that represent the change in the stock price over time. The dataset here, that we have obtained from Yahoo Finance, is a sample of the historical data of the stocks of some companies in which we use to predict the future values.

***K*eywords** Times Series · LSTM · Decision Tree Regressor · Regression

1 INTRODUCTION–

We propose to design and develop an artificial intelligence model that predicts the value of the stock market prices of IT companies, based on the analysis of time series. The various tasks involved are

1) Reading data from Yahoo Finance

2) Processing of data to make it usable for the required prediction model

3) Training model to find out the prediction with the help of decision tree regressor and LSTM allowing us to make good and efficient predictions.

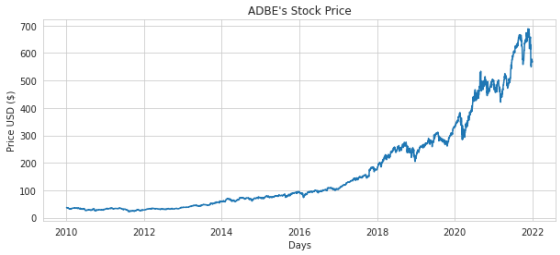
We get multiple variables from the dataset, such as High, Low, Open, Close, Volume, Adj Close. But since we predict only the future close attribute, we don't use the other variables in this research.
Processing of the data has several sub divisions like reading the data, fill the missing dates and splitting the dataset for performing tasks like training and testing to perform the required prediction.
A general conclusion closes this report by recapitulating the work carried out and presenting perspectives to our project.

2 ABOUT THE DATASET–

The dataset here is provided by the API of Yahoo Finance, which in addition to being one of the best news and media websites, provides historical stock market prices for more than 150,000 companies.
The training data consists of 537577 observations and 7 features.
For pre-processing, we fill the missing data for certain days then, we then break the data into train, validation and test sets. For each company and each date available, Yahoo Finance provides the following information:

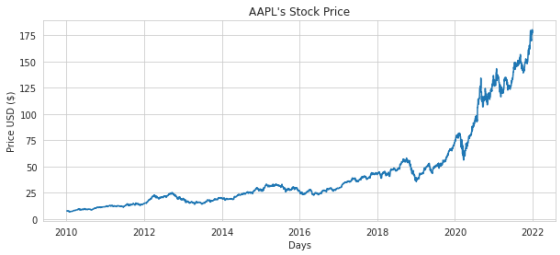
- Open: price of the stock market price of a company at the opening of the stock market.
- High: share price of a company which reached the highest level during the day.
- Low: share price of a company which reached the lowest level during the day.
- Close: price of the last sold share of a company during the day.
- Adj. Close: share price adjusted by the dividends.
- Volume: number of shares traded during the day.

For our system, we want to predict a single value of the stock price for the entire day which provides the trader with an average of the stock price over the course of a day. To do this, we base ourselves on the close price.
We have considerd 12 technology companies listed on the NASDAQ stock exchange:
APPLE (AAPL), AMAZONE (AMZN), GOOGLE (GOOGL), FACEBOOK (FB), MICROSOFT (MSFT), NVIDIA (NVDA) , NETFLIX (NFLX), TESLA (TSLA), HP (HPQ), CISCO (CSCO), IBM (IBM), ADOBE (ADBE).



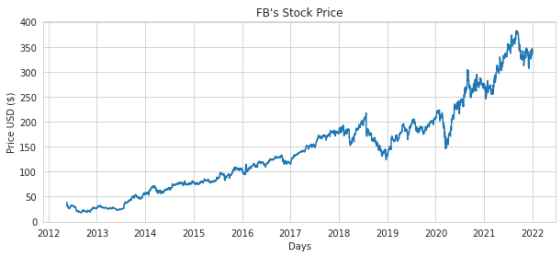
(a) ADBE's Stock Price

(b) AMZN's Stock Price



(c) APPL's Stock Price

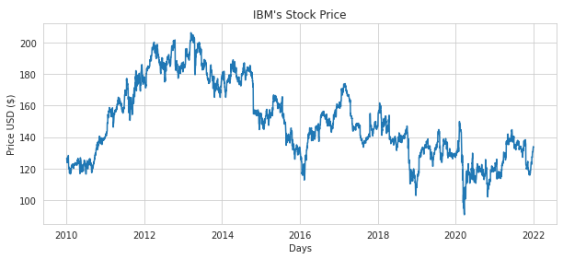
(d) CSCO's Stock Price



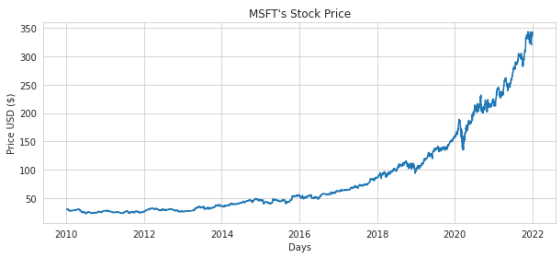
(e) FB's Stock Price

(f) GOOG's Stock Price

(g) HPQ's Stock Price

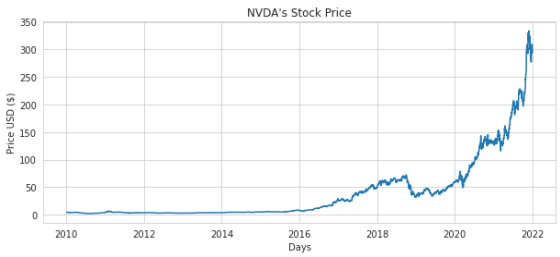


(h) IBM's Stock Price

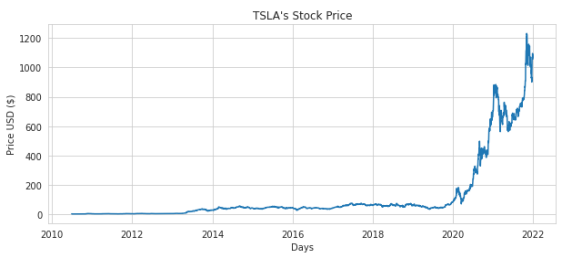


(i) MSFT's Stock Price

(j) NFLX's Stock Price



(k) NVDA's Stock Price



(l) TSLA's Stock Price

Figure 1: The Dataset

3 METHODS USED–

3.1 MACHINE LEARNING APPROACH–

We can define Machine Learning (ML) as an artificial intelligence technology that allows machines to learn without having been specifically programmed for this purpose. ML models are of two types: supervised learning and unsupervised learning.
- Supervised learning: consists of input variables (X) and an output variable (y). The algorithm allows you to learn the input to output mapping function. The goal is to predict for new input data (X) the values of the output variables (y). Supervised learning problems can be divided into two categories: regression problems and classification problems.
• Classification: A classification problem is a problem where the output variables represent categories.
• Regression: A regression problem is a problem where the output variable(s) are real values.
- Unsupervised learning: consists of having only input data (X) and no corresponding output variables (Y). The goal of unsupervised learning is to model the underlying structure or distribution in data. For our project, forecasting stock prices can be viewed as a problem of supervised regression learning. We will base ourselves on the old values in terms of time as input variables and the next 7 values in time as the output variable(s). There are various machine learning algorithms adapted to our problem, we present in what follows an algorithm among the most used in the prediction of time series, the Decision tree Regressor.

3.1.1 DECISION TREE REGRESSOR–

The decision tree is a machine learning classification algorithm that has been adapted for prediction problems. The principle consists in building a tree structure by dividing the set of input data into sets of smaller and smaller size. The following figure illustrates an example of application of this algorithm.

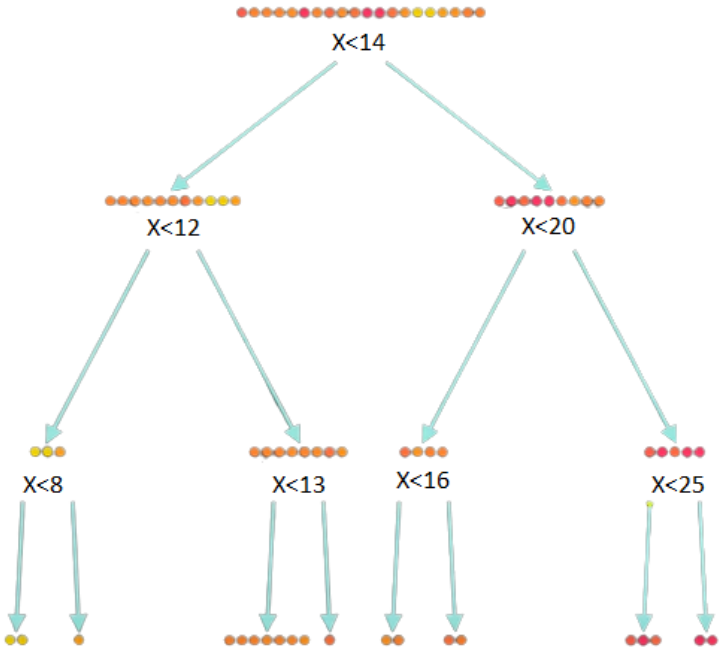


Figure 2: Decision Tree Example

The root node contains all the observations. The tree is built by splitting the nodes. A node will be divided if its impurity (the standard deviation between the data in the node) is above a given threshold, otherwise it is a leaf. To measure the quality of a separation of a node, there are various criteria among which we can cite the reduction of the variance between the values of the child nodes.

$$variance\_reduction = var(parentnode) - \sum_{i=1}^{n} w_i * var(childnode_i)$$

For this model, the predicted value (Y) is the average of the data of the leaf reached after traversing the tree according to an input X value.

3.2 DEEP LEARNING APPROACH–

Deep Learning is a sub-domain of ML inspired by neurons in the human brain. This approach attempts to identify the main characteristics of the input data to be able to predict an output result. Deep learning is based on a network of artificial neurons organized in layers of linked neurons. The data is introduced to the neural network through the first "input layer" and provides its result on the last "output layer". These two layers are connected to intermediate "hidden layers" by weighted connections as shown in the figure that follows. There are different types of neural networks but for our project, we are only interested in recurrent neural networks (RNN) because this category of neural networks is dedicated to the processing of sequences.

* Recurrent Neural Network (RNN)

An RNN is a model of the Deep Learning family adapted to sequential data predictions thanks to a memory-based architecture. It is a simple artificial network where information persists through an information feedback connection, so it becomes possible to link the information passed to the current task. The following figure illustrates an RNN associated with time t where the inputs are x(t) and its own output with the previous time y(t - 1).

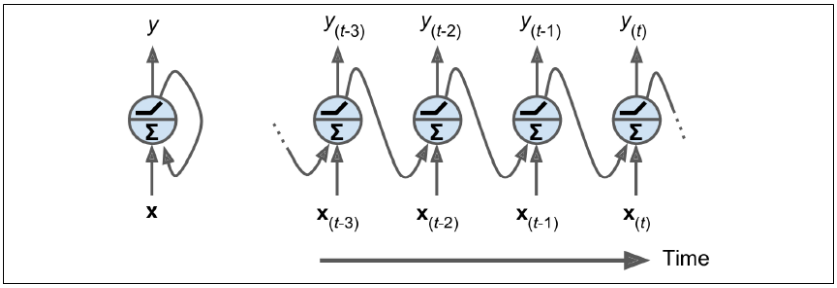


Figure 3: Recurrent neuron network

3.2.1 Long Short Term Memory (LSTM)–

The main disadvantage of RNNs is the difficulty in learning long stretches of data. Indeed, with a deep RNN architecture (with a large number of layers), the gradient becomes smaller and smaller, parameter updates become insignificant and therefore no real learning is performed. The LSTM corrects this problem by adopting internal mechanisms called gates that regulate the flow of information.

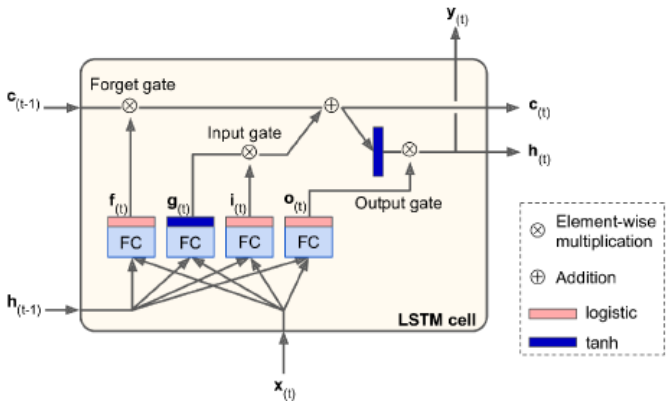


Figure 4: Long Short Term Memory

The LSTM has 3 types of gates:
- *The forget gate* (controlled by f(t)) controls which parts of the long-term state should be cleared.
- *The input gate* (controlled by i(t)) controls which parts of g(t) should be added to the long-term state.
- *The output gate* (controlled by o(t)) controls which parts of the long run the state should be read and output at this time step, both to h(t) and to y(t).

$$
\begin{aligned}
\mathbf{i}_{(t)} &= \sigma\left(\mathbf{W}_{xi}{}^{\top}\mathbf{x}_{(t)} + \mathbf{W}_{hi}{}^{\top}\mathbf{h}_{(t-1)} + \mathbf{b}_i\right) \\
\mathbf{f}_{(t)} &= \sigma\left(\mathbf{W}_{xf}{}^{\top}\mathbf{x}_{(t)} + \mathbf{W}_{hf}{}^{\top}\mathbf{h}_{(t-1)} + \mathbf{b}_f\right) \\
\mathbf{o}_{(t)} &= \sigma\left(\mathbf{W}_{xo}{}^{\top}\mathbf{x}_{(t)} + \mathbf{W}_{ho}{}^{\top}\mathbf{h}_{(t-1)} + \mathbf{b}_o\right) \\
\mathbf{g}_{(t)} &= \tanh\left(\mathbf{W}_{xg}{}^{\top}\mathbf{x}_{(t)} + \mathbf{W}_{hg}{}^{\top}\mathbf{h}_{(t-1)} + \mathbf{b}_g\right) \\
\mathbf{c}_{(t)} &= \mathbf{f}_{(t)} \otimes \mathbf{c}_{(t-1)} + \mathbf{i}_{(t)} \otimes \mathbf{g}_{(t)} \\
\mathbf{y}_{(t)} &= \mathbf{h}_{(t)} = \mathbf{o}_{(t)} \otimes \tanh\left(\mathbf{c}_{(t)}\right)
\end{aligned}
$$

Figure 5: Long Short Term Memory - Equations

3.2.2 Evaluation of prediction models

Evaluation is a crucial step in building a regression model that helps decide whether or not a model can be deployed. We present in the following some regression metrics:

– MAE (Mean absolute error): represents the average of the absolute values of the differences between the observations and the predictions of a model. It involves the first moment of the distribution.

$$mae = (\frac{1}{n})\sum_{i=1}^{n}|y_i - x_i|$$

– MSE (mean squared error): represents the mean of the squares of the differences between the observations and the predictions of a model. It involves the second moment of the distribution.

$$mse = (\frac{1}{n})\sum_{i=1}^{n}(y_i - x_i)^2$$

– RMSE (root mean squared error): is the square root of MSE

$$rmse = \sqrt{(\frac{1}{n})\sum_{i=1}^{n}(y_i - x_i)^2}$$

– Skewness = is the degree of asymmetry observed in a probability distribution that deviates from the symmetrical normal distribution. It involves the third moment of the distribution.

$$a_3 = Skewness = (\frac{1}{n})\sum_{i=1}^{n}\frac{(X_i - \bar{X})^3}{s^3}$$

– Kurtosis = refers to the degree of presence of outliers in the distribution. It involves the fourth moment of the distribution.

$$a_4 = Kurtosis = (\frac{1}{n})\sum_{i=1}^{n}\frac{(X_i - \bar{X})^4}{s^4}$$

4 EXPLANATION OF THE WORK DONE–

4.1 APPLICATION OF THE MACHINE LEARNING APPROACH: DECISION TREE

- Preprocessing of the dataset:
We adopt the sliding window method which allows the model to take into account several previous steps in its future prediction.
This prediction model makes it possible to predict a single value (that of day d + 1) based on several past values (the number of values taken into consideration represents the window size = 63 in our case).
- Model training:
Application of the decision tree algorithm on the training dataset.
- Model rating:
The analysis of the following figure shows that the predictions of the decision tree models are more or less satisfactory for HPQ, CSCO and IBM only.

	mae	rmse
AAPL	12.10	16.84
AMZN	68.22	93.24
GOOG	688.73	778.62
FB	46.70	55.44
MSFT	53.41	64.60
HPQ	4.82	5.75
NVDA	57.75	80.69
NFLX	36.37	57.04
CSCO	0.77	1.25
ADBE	68.00	91.04
IBM	1.27	1.83
TSLA	134.19	198.69

Figure 6: Evaluation of Decision Tree model

4.2 APPLICATION OF THE DEEP LEARNING APPROACH: RNN-LSTM

In this type of network, we only consider the stock price in the prediction. We use the "many-to-one" architecture of recurrent neural networks for this model. This type of network takes several values as parameters to produce a single one.

- Preprocessing of the dataset:

We carried out the same processing of the dataset as before since we have the same input and the same output.

- Choice of an architecture for the neural network:

We tested five models on our dataset with different values for the hyperparameters (window size, number of neurons per layer ...) and different number of layers.

The following figure shows the model we applied.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 LSTM (LSTM)                 (None, 10)                480

 Output (Dense)              (None, 1)                 11

=================================================================
Total params: 491
Trainable params: 491
Non-trainable params: 0
_________________________________________________________________
```

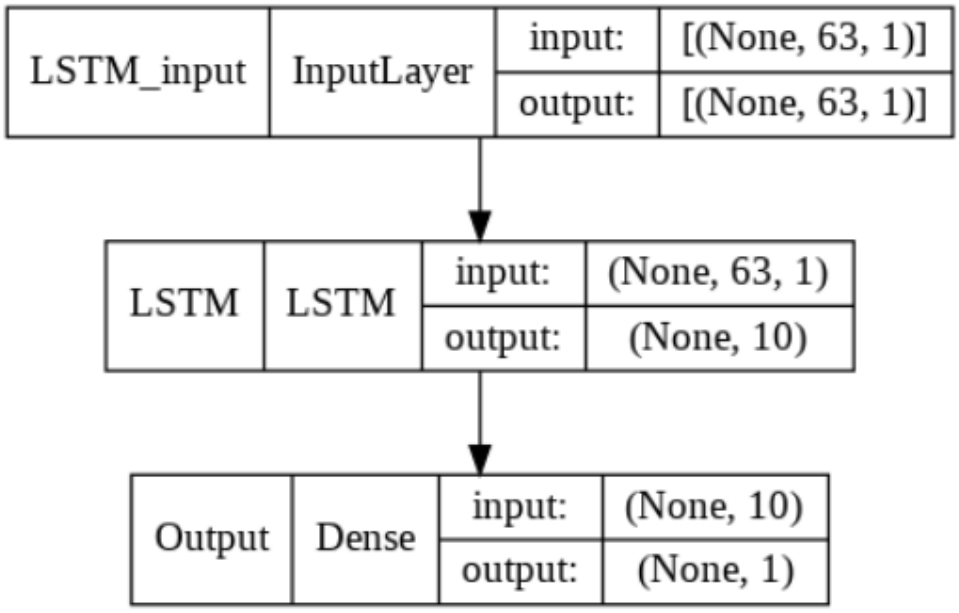


Figure 7: Architecture of the RNN-LSTM model

This network is made up of 3 layers. The first and second layers are layers of LSTM neurons. The third and final layer of the network, the Dense layer is used to produce predictions.

- Model compilation:
This phase is important since it allows you to specify:
• model optimizer: "ADAM"
• loss: the model tries to minimize this function of evaluating errors during training.
• a metric for measuring the error: MAE & RMSE
- Learning the model:
In this phase, we use the principle of "EarlyStopping" which makes it possible to stop the learning of the model before it reaches an overfitting state and to accelerate the model training phase.
- Model rating:
The analysis of the following figure shows that the predictions of the decision tree models are more or less satisfactory for AAPL, FB, MSFT, HPQ, CSCO and IBM.

	mae	rmse
AAPL	3.80	5.32
AMZN	49.99	61.58
GOOG	191.31	215.17
FB	5.90	7.40
MSFT	6.12	7.85
HPQ	1.52	1.66
NVDA	20.46	30.25
NFLX	10.83	14.51
CSCO	0.52	0.69
ADBE	17.25	21.35
IBM	1.25	1.84
TSLA	22.91	30.70

Figure 8: Evaluation of RNN-LSTM model

5 RESULTS–

Our project link is :

https://github.com/IhebMarnaoui/
PREDICTING-THE-STOCK-PRICE-OF-COMPANIES-USING-MACHINE-LEARNING-MODELS

Since our dataset is composed of different companies, and for simplicity of explanation, we are going to study the case of CSCO company's stock in what follows since it presents the minimum values of errors in both models. The rest of the results and figures are available via the project link above.

5.1 FIGURES-

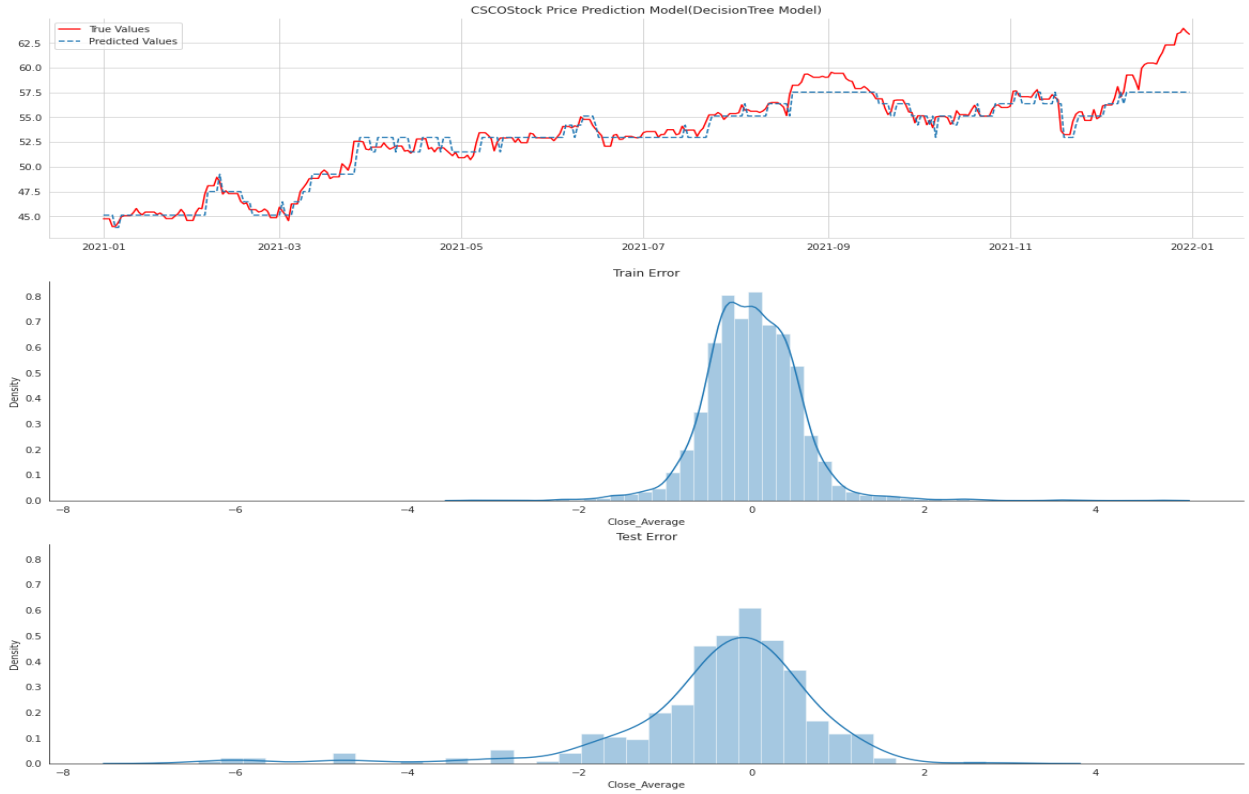


Figure 9: CSCO Stock Price Prediction Model (Decision Tree Model)

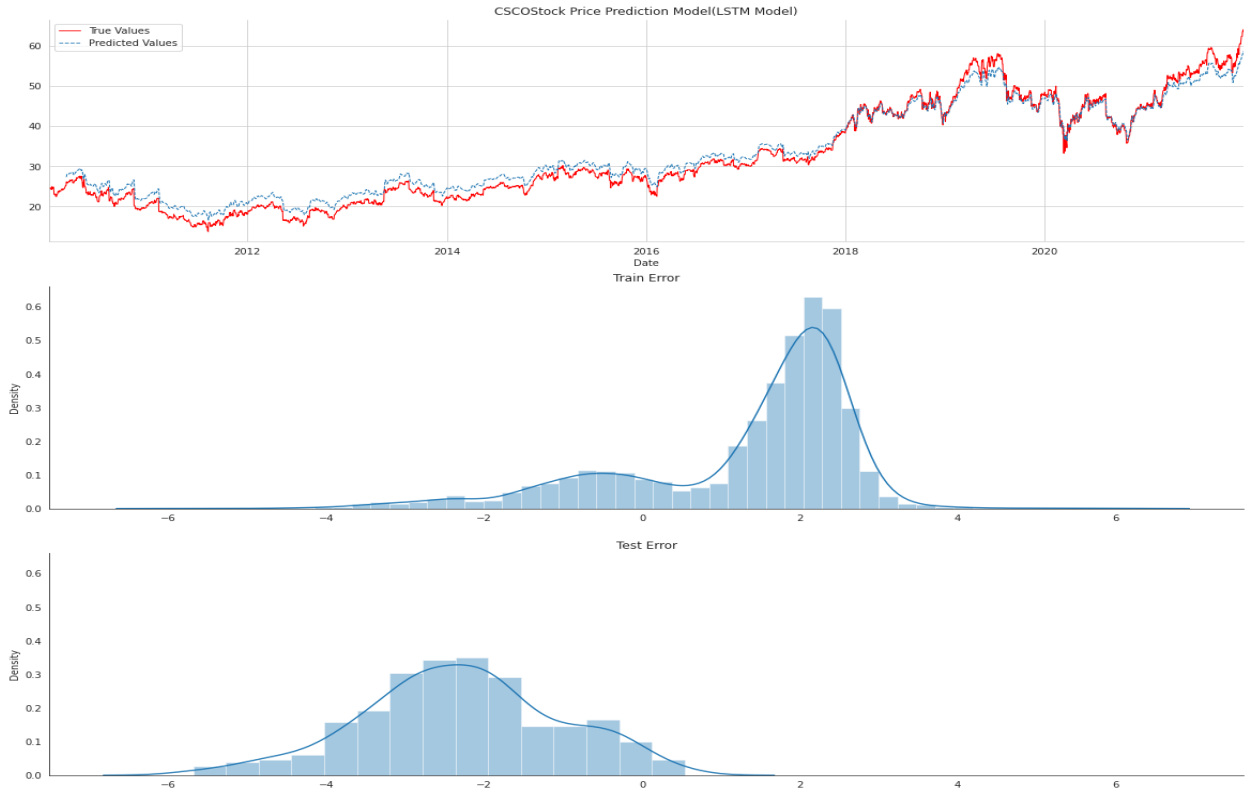


Figure 10: CSCO Stock Price Prediction Model (LSTM Model)

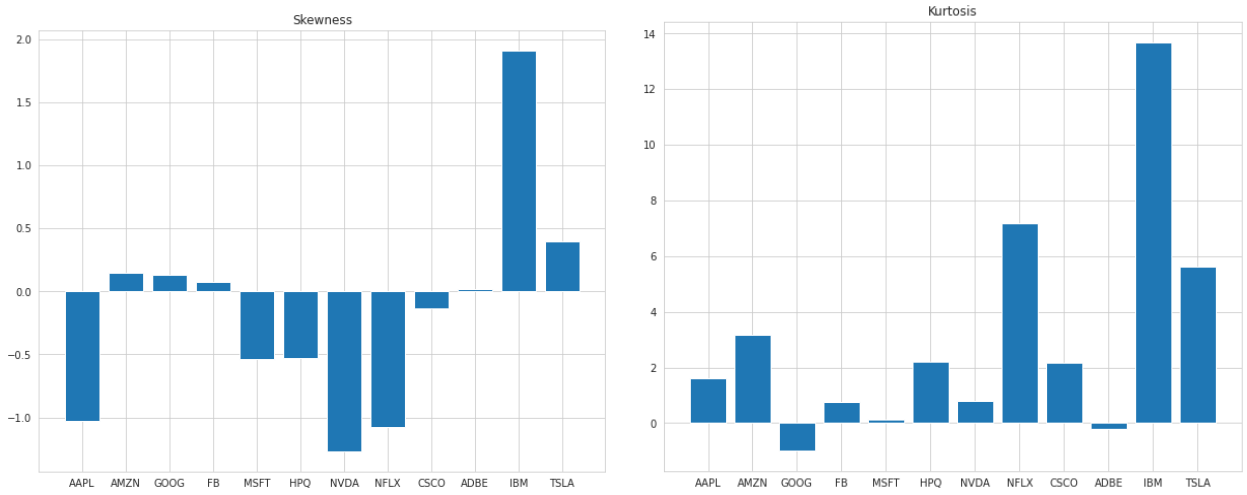


Figure 11: Skewness & Kurtosis measures (LSTM Model)

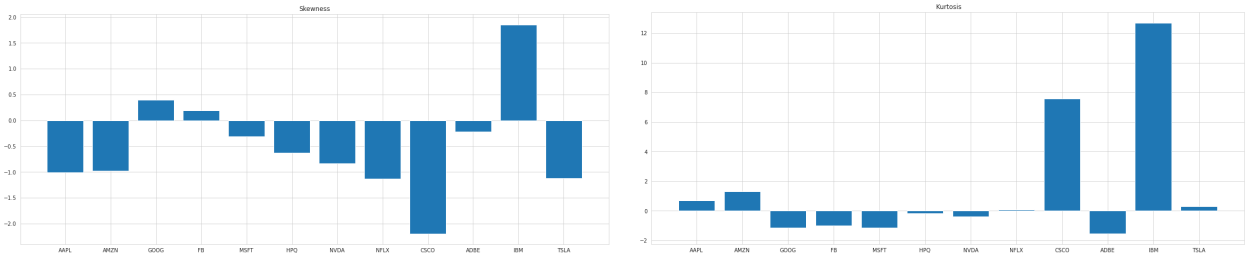


Figure 12: Skewness & Kurtosis measures (Decision Tree Model)

6 DISCUSSIONS–

As already stated before, we applied Decision Tree Regressor and LSTM from which we got several values that can be used in the analysis and prediction, them being MAE, RMSE, Kurtosis and Skewness. These values are different for Decision Tree Regressor and LSTM and are given in the table below:

Table 1: Decision Tree Regressor VS LSTM (CSCO's Stock Price)
(MAE , RMSE , KURTOSIS AND SKEWNESS)

	MAE	RMSE	KURTOSIS	SKEWNESS
LSTM	0.52	0.69	-0.18	-0.09
Decision Tree Regressor	0.77	1.25	7.59	-2.20

7 CONCLUSIONS–

After having applied both Decision Tree Regressor and LSTM models, we observe that LSTM worked better than Decision Tree Regressor in terms of prediction efficiency and accuracy.

Significant analysis was made with the help of Root Mean Squared Error, Mean Absolute Error, Kurtosis and Skewness. In this project, we tried to come up with the best model for the prediction of Stock Prices.

Another thing that we observed is that the closer the distribution of the errors in the histogram to 0, the minimum the errors are. The accuracy we acquired is creditable considering the fact that the dataset was large with several errors and hidden values. We must also consider that all the variables which are there in the dataset are not strongly co-related with the close variable.

8 ACKNOWLEDGEMENT–

We would like to thank our mentors Mr. Faouzi Ghorbel and Mrs. Mouna Ben Salah Baklouti, for their invaluable advices, their availability and their supervision which allowed us to carry out our project.

References

[1] Binoy Nair • V. P. Mohandas • N. R. Sakthivel A Decision Tree- Rough Set Hybrid System for Stock Market Trend Prediction

[2] https://www.kaggle.com/faressayah/stock-market-analysis-prediction-using-lstm

[3] Osman Hegazy • Omar S. Soliman • Mustafa Abdul Salam A Machine Learning Model for Stock Market Prediction

[4] https://www.spcforexcel.com/knowledge/basic-statistics/are-skewness-and-kurtosis-useful-statistics