“To use AI to predict data from the stock market and measure the impact of different models and parameters affected at the stock market.”

Author

Smith R Dbritto

Student ID

M00689896

Supervisor

Mr. Ahmed Eissa

Module Leader

Dr. Can Başkent

BSc Computer Science
Middlesex University London

Contents

1.Declaration

I, Smith Rajesh Dbritto, hereby confirm that the work presented here in this report and all other associated material is wholly my work. The information derived from the literature has been duly accredited in the text and a list of references is provided. No part of this dissertation was previously presented for another degree or diploma at this or any other institution.

SMITH RAJESH DBRITTO

Signature Smith Rajesh Dbritto Date: 20/04/2023

2.Acknowledgement

I would like to acknowledge and give my warmest thanks to Mr. Ahmed Eissa my supervisor for allowing me to work on this project. His guidance and support carried me through all stages of the project.

As well as gaining valuable knowledge on AI and exploring new helpful algorithms. I would also like to extend my gratitude to the module leader Dr Can Başkent for providing continuous support through lectures and workshops.

3. Abstract

For stock market prediction this study initially provides information on how prediction is made, further, upon re-iteration, the topic has been updated to recording and displaying the impact of the parameters that is generally caused due to the margin of error. After inspecting and deciding among multiple models such as ARIMA, ANN, GAN, and such we end up on LTSM and further to the code.

In the initial steps, we go through the process of checking different data sets to predict the next day's data based on the previous 60 days collected data. Furthermore, after analysis we research different methods for efficiency, discuss other such impacts caused by the margin of error.

In the end after considerable evaluation and discussions, we were able to conclude that aside from few of the great crisis of 2008 and recent issues, there have not been any clear indication of recordings aside from a handful of papers. This paper being another clearer and more efficient attempt as the paper for the above's cause. Multiple sources and information has been collected and recorded regarding the positive and negative affects of parameters impacting the stock market.

4. Introduction

The phrase "stock market" refers to a variety of marketplaces where shares of publicly listed companies can be purchased and sold.

When stocks are sold the value of it decreases causing a dip, and when bought value increases. However, in this case, we are focusing more on the aspect of what we can accomplish with this information. At times, the stock market is affected by certain parameters. These parameters are irregular, and we cannot predict the future based on the data we know about now possibly due to insufficient data incalculable by man. This is visibly evident in an example we can look at.

To summarize a recent event on Twitter, Elon Musk recently changed the verification process to get a checkmark to visualize an account from being official, until they made an immediate change, people could pay 8$ per month and get the verified checkmark on their Twitter account, and with this, came newer problems as you can imagine. Someone had created a fake Eli Lilly (a pharmaceutical company) account, and tweeted “We are excited to announce insulin is free now.” Due to this, everyone started to withdraw their stocks from the company to avoid losing any money and in turn dropped the value of the company by a tremendous amount.

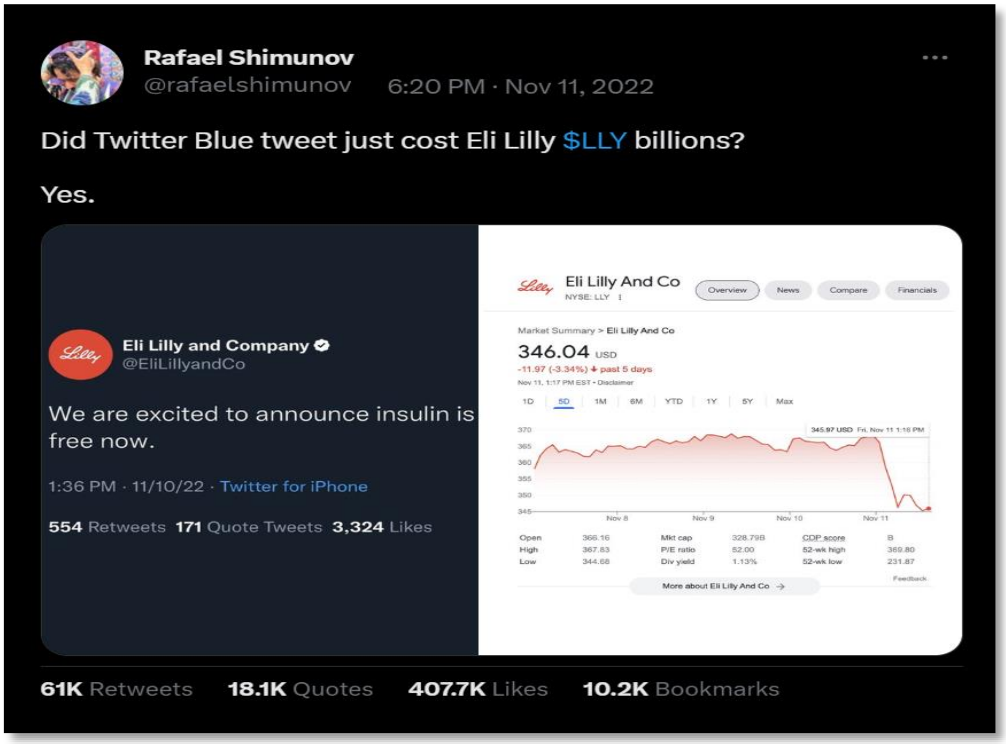

Figure 1.0: Screenshot of the twitter page indicating the False account.

5. Problem Statement

In this current state, we have progressed past creating an AI that is able to predict the future values for stocks, however the margin of error is far greater. Hence this project aims to further research data regarding the effects of each of the parameters, its importance, and if there exist any possible methods to reduce the margin of error to predict accurate values.

The report will primarily describe the current state of the AI model, on what results we are able to achieve. Unlike other papers, my goal is to find out the impact made by the following: Human (ie. impact of the fake company account or the actions made by leaders of a country to resolve or start a conflict) or nature's impact (Diseases or possibly COVID 19.) on the stock market.

6. Literature Review

6.1 Time Series

Time series is a collection of data which varies across time. This data is required for our knowledge to import it into the AI however, there are multiple parameters and variables we need to learn before beginning with importing. This type of data are generally statistics plotted into a graph such as number of items in an inventory sold across different times of the year or in cricket in the first over, there will be 10 runs, in the next over it would be 15 runs then it would progressively change depending on the team playing.

Figure 1.1 (Examples of Time series data)

Time series can be broken down into multiple components which are also the parameters which affect the data we value.

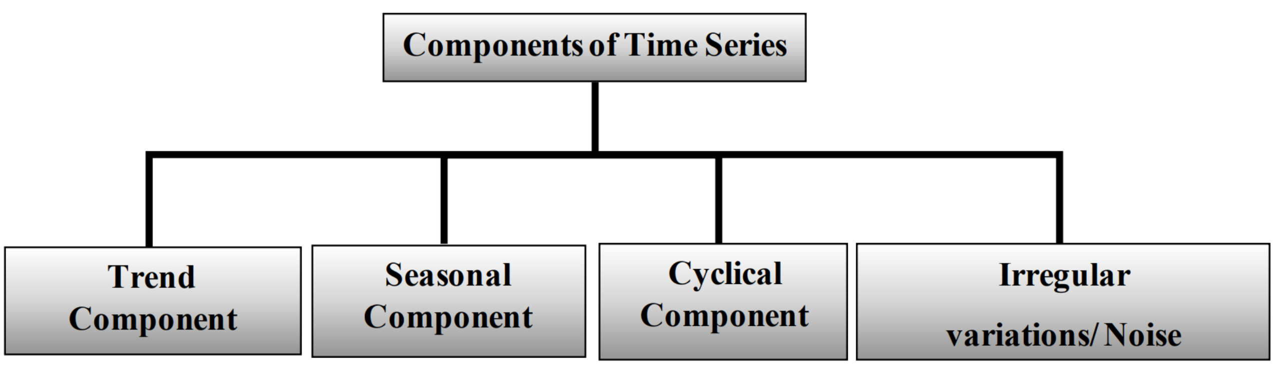

Figure 1.2 (Image regarding Components of Trend Series)

Trend Component is one of the components which helps us in the analysis of the overall direction of the graph (upwards or downwards).

Figure 1.3 (An increase in trend for bitcoin in the past 5 days.)

Figure 1.4 (An overall decrease in trend for bitcoin in the past 1 year.)

In Figure 1.3 we can see for bitcoin the trend for a 5-day time period is increasing.

Whereas if we were to compare it with Figure 1.4 we see a more clearer picture of the overall trend. Next component we will be covering is the Seasonal Component. Unlike long term trends the seasonal component limits itself from about 6 months to a year. It is often monthly or quarterly and consists of short-term regular wave like patterns.

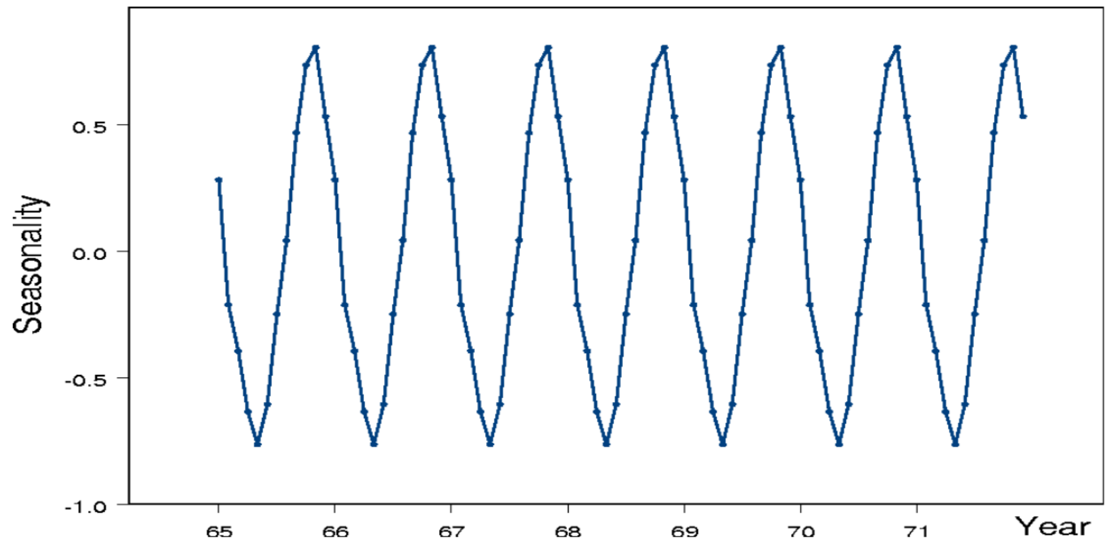

Figure 1.5 (Depiction of Seasonal Component)

A Cyclical component is a sequence of phases of expansion and contraction associated with a trend, excluding the irregular component. It also informs us about in the long term (roughly 10- or 20-years' time), in which way the stock peaks and dips. It is often measured from peak to peak or trough to trough.

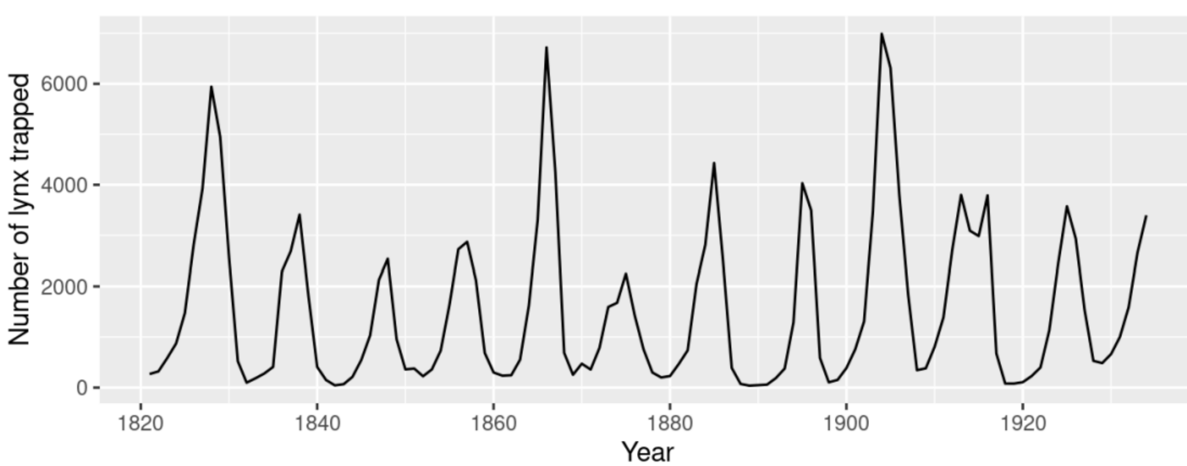

Figure 1.6 (Depiction of Cyclical Component)

If it is possible to predict each of these trends the accuracy of our predictions will be almost if not identical to the original. However, we must take another component into consideration whilst examining all the information. This component is called Noise. Noise is the occurrence of irregularities in the trend. It goes by many names such as irregular or residual component. When we look at traditional methods of looking at stock market data or a time series data for prediction, the key idea is to primarily make the time series stationary.

Since time series contains multiple components as discussed earlier, it is difficult to incorporate each one of the features to make a successful prediction. Hence, we start upon Stationary Time Series also known as Stationarity:

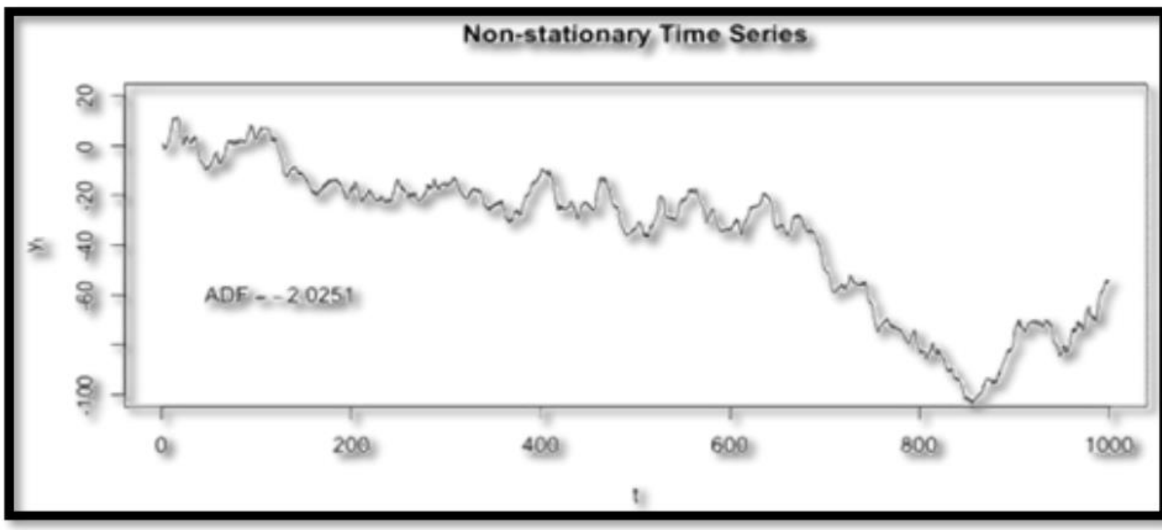

Figure 1.7 (Non-Stationary Time Series Graph)

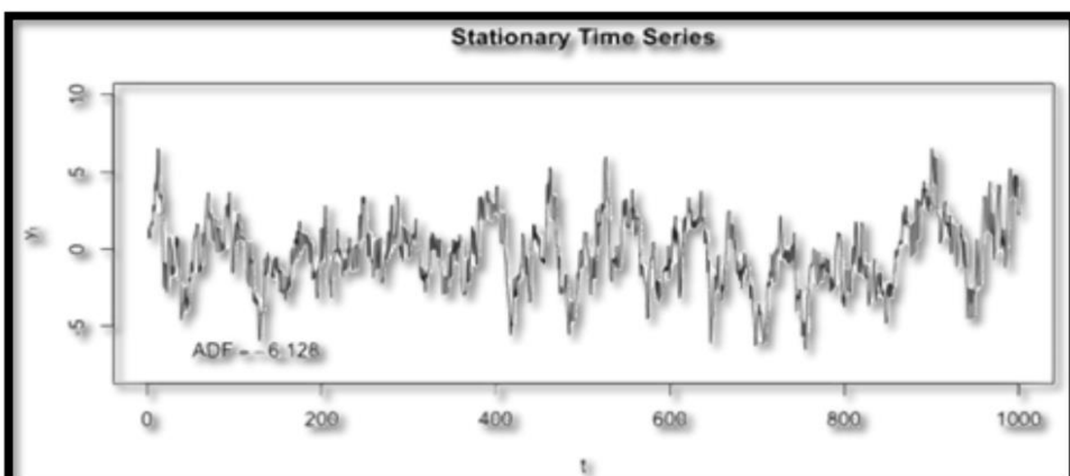

Figure 1.8 (Stationary Time Series Graph)

The way to implement using the traditional method would be to follow the loop in the flow chart illustrated below:

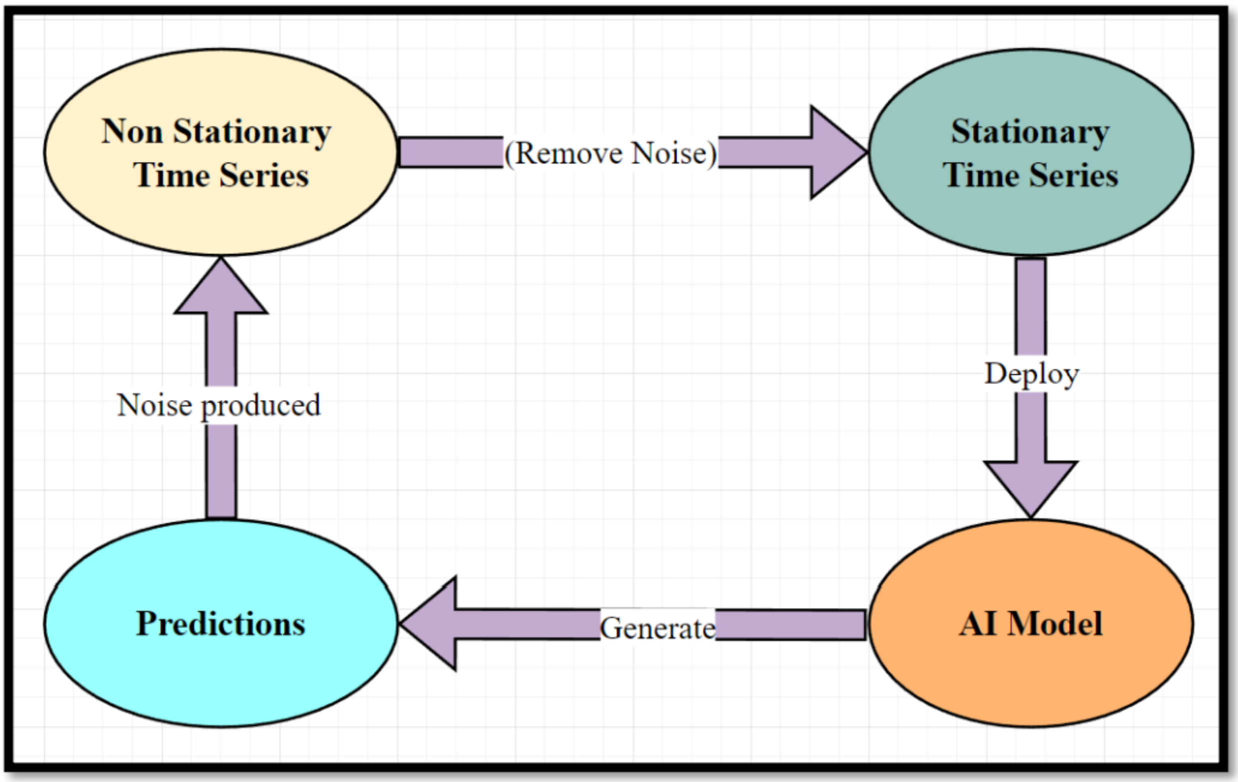

Figure 1.9: Flowchart illustration

By looking at Figure 1.9 we can understand the standard approach to process time series and is listed as follows:

- First one must convert any type of non-stationary time series to a stationary time series then one can model it and utilize their AI model.
- Once the AI has made predictions, then it can be converted back to nonstationary. The noise that was initially removed is later generated. That's the traditional approach.
- But this process causes some inaccuracies, so it is unviable. Hence a stationary time series is defined by two things, a constant mean, and a constant variance.
- If the mean value of the price over a period remains constant, the time series is stationary.
- If a mean value changes with time it is a non-stationary time series. Time series of current stock is always non-stationary, so the use of AI directly is not

feasible. This change is due to false assumptions about underlying traditional methods.

- The AI can accurately function if it assumes the time series is stationary.
- Therefore, the time series must be converted to stationary, after which factors such as trends, seasonality and cyclicity can be added in. After the algorithm is completed, the noise can be added back in.
- There are two tests to determine stationarity of data; Rolling Statistics in which one plots the moving average or variance and see if it varies with time and the ADF Test (Augmented Dickey-Fuller Test) which is a null hypothesis is that the time series is non-stationary. The test results comprise of test statistics and some critical values.

Methods to make the timeseries stationary include:

- **De-trending:** It includes regressing against covariates other than time
- **Seasonal adjustment:** It incorporates differences however it could be viewed as a different mechanism.
- **Transformation of the data:** It alters a difference operator inherently into another.
- **EDA smoothing methodologies**, (such as expelling a shifting median), can be regarded as non-parametric detrending approaches.
- **Moving Average Smoothing** is a major technique used to make a stationary timeseries.
- In order to see the overall pattern of movement over time, **Calculate** the moving averages
- The **Average** of consecutive time series values over a given length of time

$$S_t = \frac{(Xt{-}k + Xt{-}k{+}1 + Xt{-}k{+}2 + \ldots + Xt)}{k}$$

Figure 1.10: Formula processed for calculation.

Based on the average of the last k elements of the series in MAS time series

- The result is more variable when k is smaller
- When the value of k is larger, smoothness is increased.

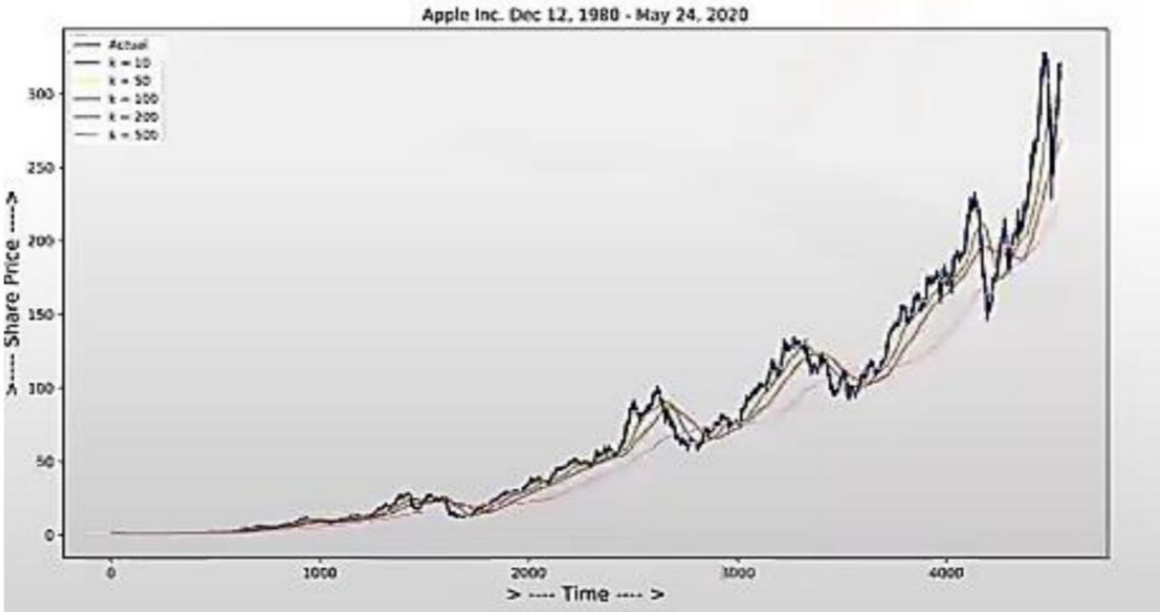

Figure 1.10.1: Smoothing of data.

The data is smoothed out by a 5-year moving average, that makes it easier to detect the underlying trend.

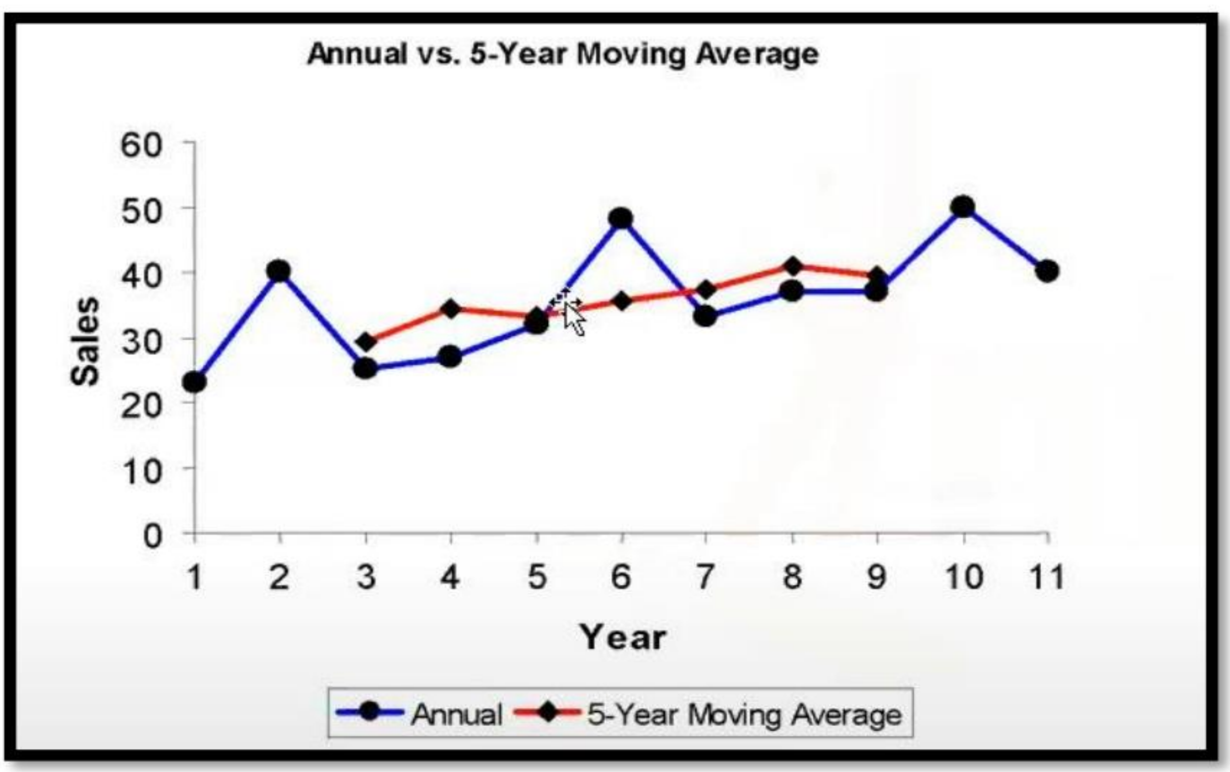

Figure 1.11: Annual vs. 5 Year Moving Average

7. Requirements Specifications

All the requirements listed below are free and publicly available.

- To be able to research the time data papers as well as paperswithcode.com/task/time-series-forecasting for the test papers for comparing different AI models.

- GitHub Desktop will be required for code management and publication.
- An IDE (Visual Studio Code used) to run the program.
- Kaggle for additional data sets
- Access to the Yahoo finance API

Name	**EpitaphAI - An AI that can predict the next trend in the stock market**
Description	**Past data sets should be imported from the API of relevant websites, and the application will analyze and predict the near future.**
Scenario	**An investor is about to invest his capital, but is reluctant in making his decision as there is too much noise (data is irregular) in the data and is unable to make a decisive decision. Thus, EpitaphAI is referred to perform multiple predictions and with sufficient information, the person can finally make the right decision.**
Precondition	**The user must have Jupyter to install and run the code.**
Trigger	**In order to begin the prediction process, the user must click on 'Predict' after the program has been run.**
Basic Flow	**Upon starting the program, the software will import the dataset from the API, start analyzing the data, and then make predictions based on the generated time series.**

Figure 1.12 (Use case 1: User initiates the program to begin the prediction process)

8. Design and Analysis

8.1 App UI Design

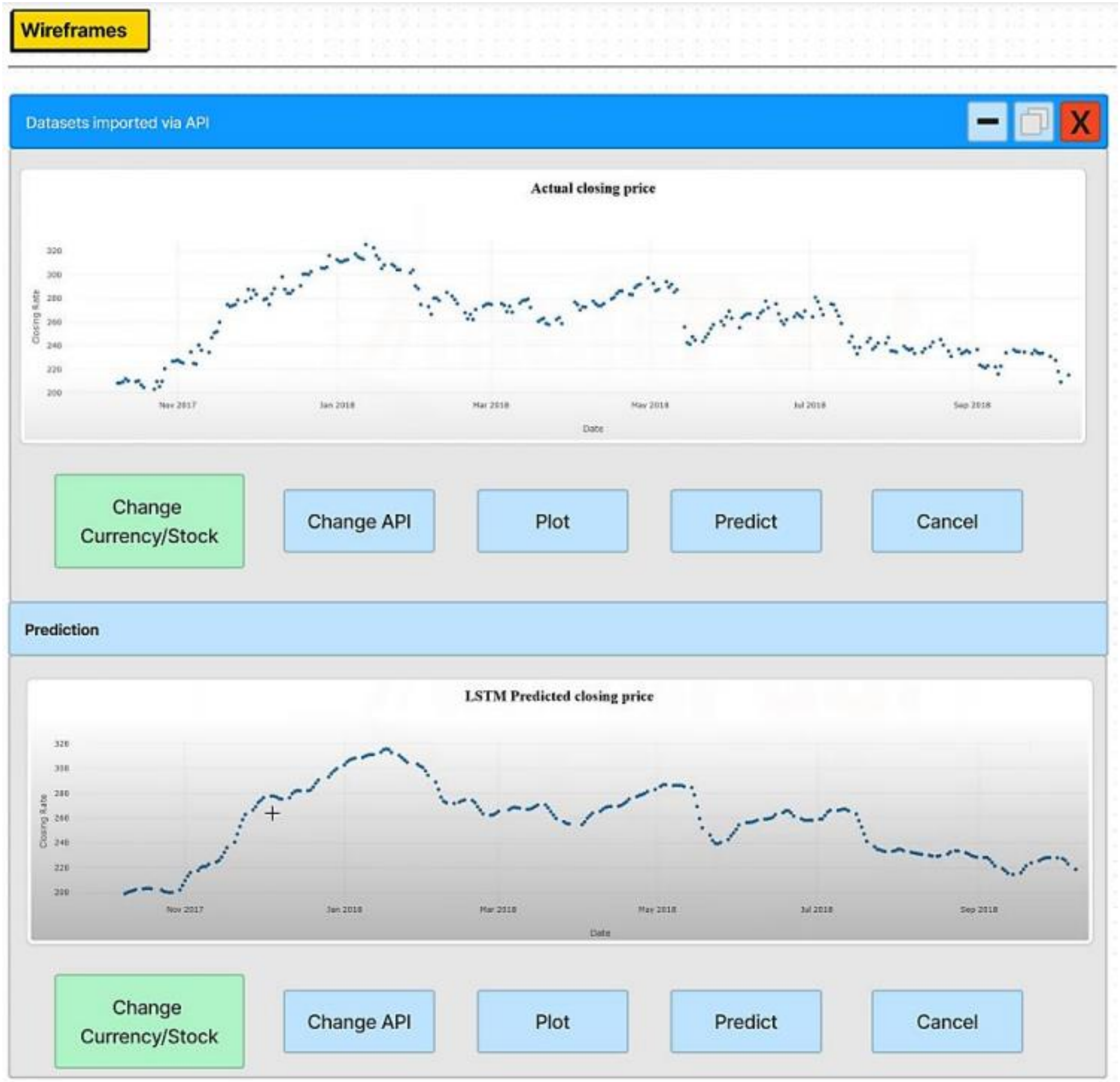

Figure 2.0 (Wireframe /Mockup of the Application.)

The Wireframe in figure 2.0 was created using Figma.

Artificial intelligence has recently been applied to address the chaotic and random time series data. The intense computational use of intelligent predictive models has commonly been studied under machine learning. Compared to the more traditional models, machine learning models provide more flexibility, do not require distributional assumptions, and can easily combine individual classifiers to reduce variance [29].

After reviewing each study, they can be briefed as follows:

Study by *Jingyi Shen and M. Omair Shafiq* in 2020.

They have collected and processed over 2 years of Chinese stock market data and have predicted price trends using a Long short-term memory (LTSM) based model.

“They applied the feature expansion (FE) approaches with recursive feature elimination (RFE), followed by principal component analysis (PCA), to build a feature engineering procedure that is both effective and efficient.” [2]

Two other popular machine learning methods, multilayer perceptron network and long short-term memory neural networks, are applied to the pattern recognition framework to evaluate the dependency of the prediction model.[31]

8.2 LTSM

LTSM, also known as long short-term memory network is used in deep learning and is a variety of Recurrent Neural Networks (RNNs) that can solve long term dependencies, in sequence prediction problems.

In 1997, Hochreiter and Schmidhuber proposed LSTM, which had achieved surprising performance in the NLP field [34]. LSTM aims at resolving long-term dependence problems based on improved RNN (Annotation) neural network. Keeping information in mind for a long time is an inherent characteristic of LSTM [35].

All RNN models have a chain form of repetitive neural network modules. As shown in Fig. 2.1 about the standard RNN models, this repetitive module is usually straightforward in structure, such as a tanh layer. [45]

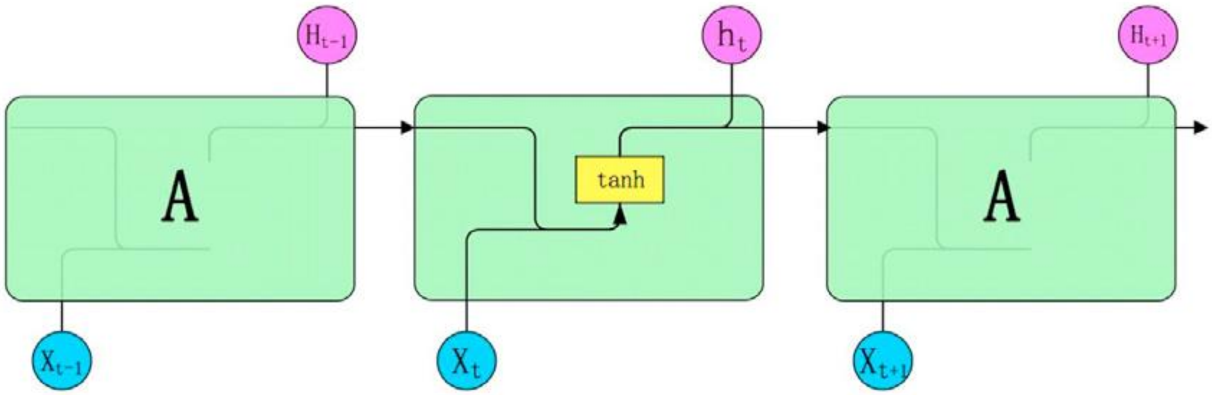

Fig. 2.1. Standard RNN Model Structure

Similarly, as a variant of RNN, LSTM also has this chain module structure, shown in Fig.2, but with different repetitive modules and layers. As Fig.2.1 shown, LSTM has three more gates than RNN with only the tanh layer.

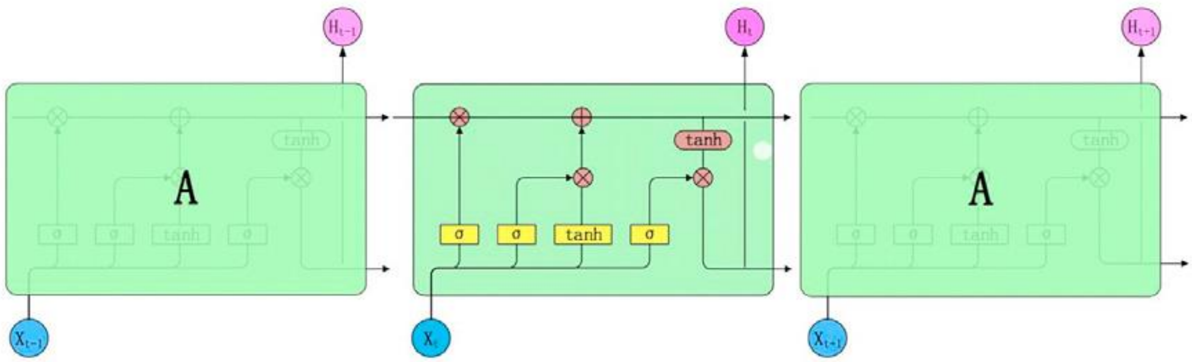

Fig. 2.1.1 LSTM Model Structure [34]

The key to LSTM is the cell state, which is the horizontal line that runs through the top of the chart. LSTM deletes or adds information to the cell state. This alteration changes the structure of cell state information, which are gates. Conclusively, LSTM has three gates.

8.2.1. Forgotten Gate

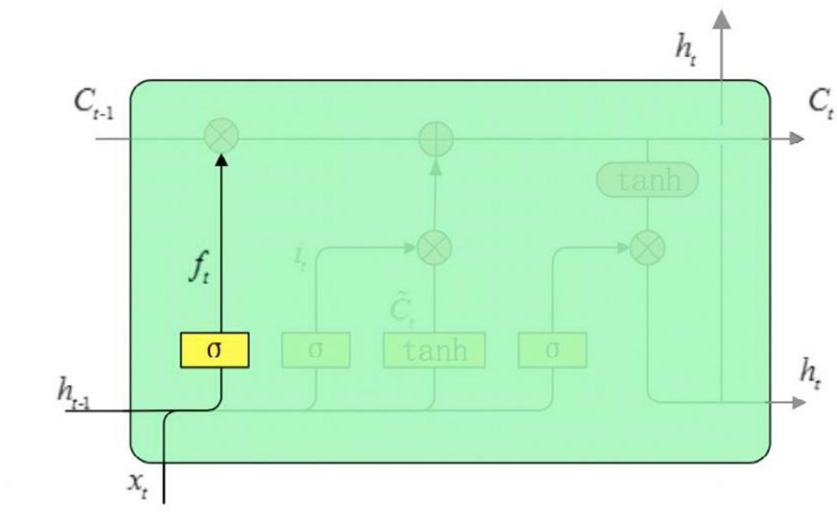

Fig. 2.1.2. Forgotten Gate

LSTM processes sequential data from left to right. In the face of a large amount of miscellaneous information, the forgetting gate decides which information of cell status is lost. According to $ht-1$ and xt, the forgetting gate calculates $ft \in 0,1$ as the input of state $Ct-1$. $ft = 0$ is for "total discarding" meaning all discarded and $ft = 1$ for "total acceptance" meaning all accepted. ft evolves as follows:

$$ft = \sigma(Wf * [ht-1, xt] + bf) \quad \ldots\ldots\ldots\ldots\ldots\ldots(1)$$

where Wf and bf represents the weight and error of forgetting gate, respectively.

8.2.2 Input Gate

The input gate consists of two parts: the first part is a sigmoid function, the output(it)is to decide which value to update; the second part is a tanh activation function, the output is C t. The multiplied results of it and C t are used to update the cell state. The formulas are as follows:

$$it = \sigma(W_i * [ht-1, xt] + b_i), (2)$$

$$C\,t = \tanh(W_C * [ht-1, xt] + b_C), (3)$$

Figure 2.1.3: Formula

where Wi and Wc represent the corresponding weight and bi, bc present the error.

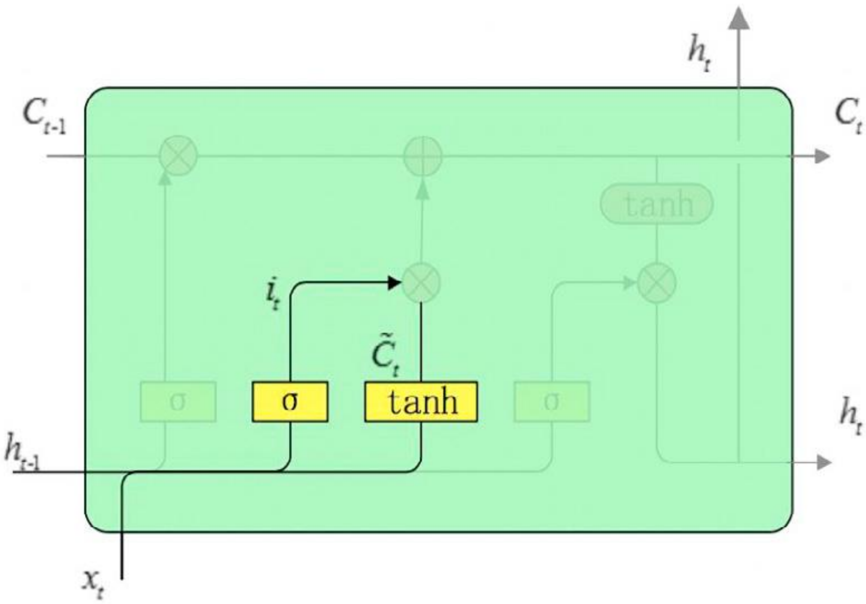

Fig. 2.1.4: Input Gate

Combining the forgotten gate and the input gate, the cell status is updated to Fig. 5:

$$\mathbf{C_t = f_t * C_{t-1} + i_t * C_t \ (4)}$$

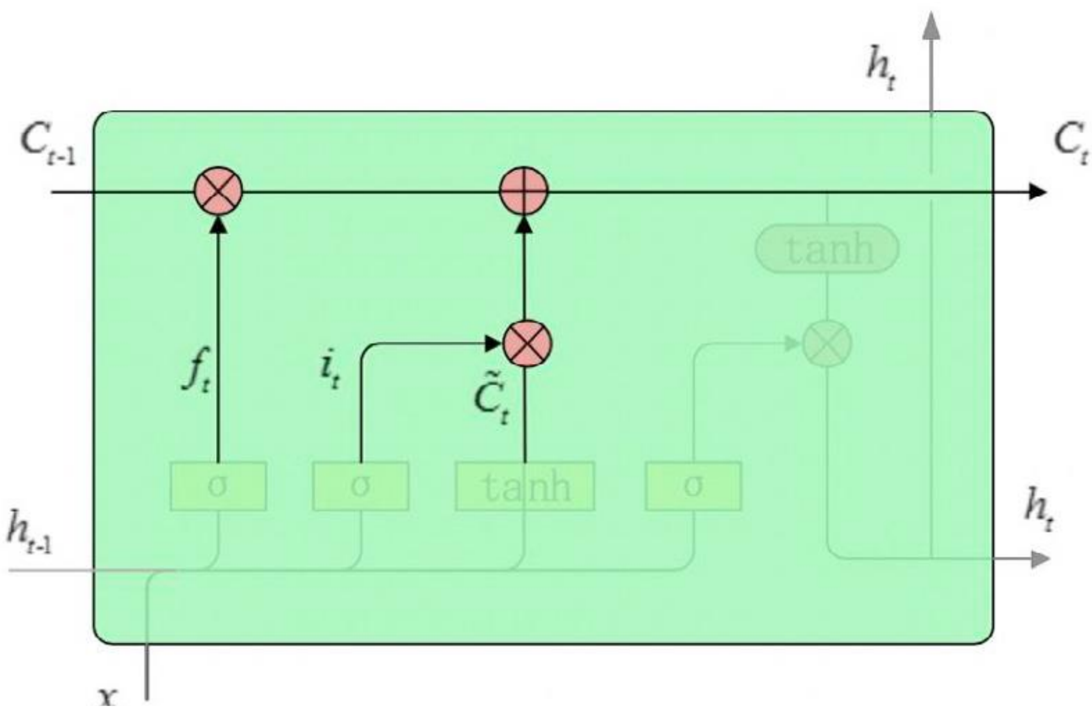

Fig. 2.1.5: Updated cell status

8.2.3 Output Gate

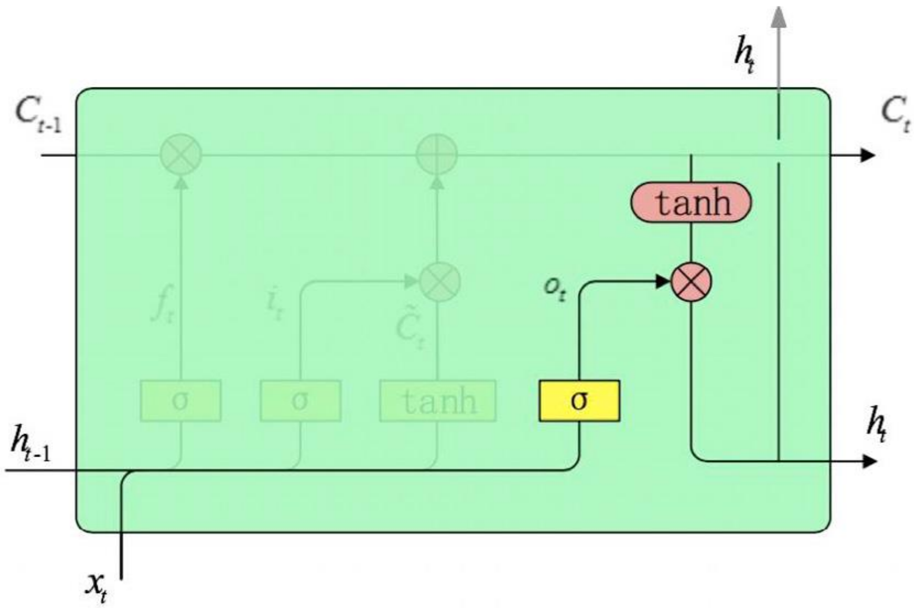

Figure 2.1.6: Output Gate

After forgetting the gate and the input gate and completing the cell status update, the output gate determines the output information. The first layer is the sigmoid layer, which determines which parts of the output cell state, and the second layer is a tanh function, which deals with the results of the first layer. Then, the outputs of the two layers o_t and tanh are multiplied to determine the final results.[45]

$$o_t = \sigma(W_O * \ [h_{t-1}, x_t] + b_O), (5)$$

$$h_t = o_t * \ \tanh(C_t), (6)$$

Figure 2.1.7: Equation 5 and 6

where Wo and b_o represent the weight and deviation of the output gate respectively, and h_t is the final output value.

8.2.4 Dropout

Dropout is a regularization technique of neural network model proposed by Srivastava et al [36]. Dropout is a technique of randomly ignoring neurons in the training process. They randomly "Dropout" means that the contribution to the activation of downstream neurons is temporarily eliminated, and no weight updates will occur in these neurons. When the neural network is trained, different neurons will adjust their parameters according to different characteristics. Neurons become dependent on this feature, some of which are even harmful. Learning too many of these features may lead to the decline of generalization ability of the model and cannot adapt to data other than training data. The formula is as follows: The formula is as follows:

$$y = f(W * \ d(x)) \quad (7)$$

$$d(x) = \begin{cases} mask * x, training \\ (1 - \mathrm{p})\mathrm{x}, \mathrm{else} \end{cases} \qquad (8)$$

Figure 2.1.8: Equation 7 and 8

Among them, p is the dropout library, and mask $1 - p$ is the binary vector produced by Bayesian effort distribution with probability. Dropout does not modify the cost function, but the deep network itself. Dropout randomly "deletes" some hidden neurons in the network, keeping the input and output neurons unchanged. In this way, for a network, dropout is like training several different neural networks with the same data, resulting in several different degrees of fitting state. But these networks have a common loss function, which is equivalent to optimizing the neural network itself and getting the average value of all states. At the same time, it reduces the synergistic relationship between the neural units and increases the robustness of the network.[37]

8.2.5 Deep Bidirectional LSTM

Deep Bidirectional Long Short-Term Memory (DBLSTM) is isolated into an input layer, a bidirectional LSTM section, a full association portion and a yield layer. The bidirectional LSTM portion is composed of multilayer bidirectional LSTM, and the complete association portion is composed of a multilayer full association layer. [45]

DBLSTM has the following advantages:

1) It can avoid RNN gradient disappearance and gradient explosion in long time series.
2) It can learn time-dependent information.
3) It can make use of the context relationship of forward and backward time series.

To realize profound information mining, multiple bidirectional LSTM layers are superimposed to memorize the profound highlights of time arrangement through multilayer neural network structure. In expansion to including a bidirectional LSTM layer, you'll moreover include completely associated layers (FC). [45]

FC has great nonlinear mapping capacity and can weigh the nonlinear characteristics of bidirectional LSTM yield, that's, to com- bine these nonlinear highlights. This preparation is fundamental to memorize a (nonlinear) condition in a vector space, and to memorize these nonlinear combinatorial highlights in a basic way of weight.

Be that as it may, with the increment of the organize layer, the preparing trouble of the show increments, the joining speed moderates down, and the issue of over-fitting is simple to happen. So, Dropout procedure is utilized to fathom these issues.

The guideline of Dropout is naturally to halt the yield of neurons with pre-set likelihood when preparing organize. [45]

The "strike" of a few neurons implies that as it were portion of the information highlights is included within the preparing of each arrange, to anticipate the arrange from learning as well much of the information highlights of the preparing set and accomplish the reason of anticipating over-fitting.

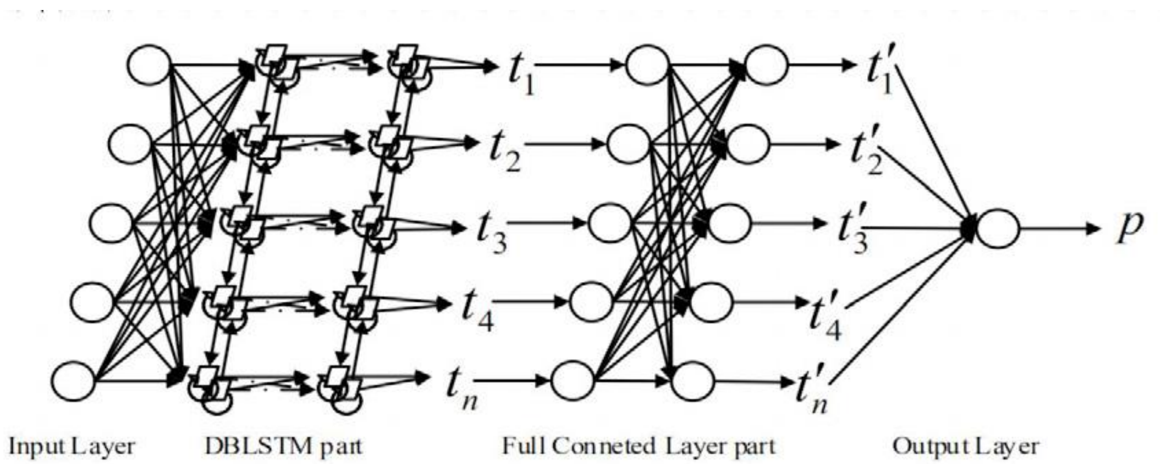

Figure 2.2: Referencing the information above

For further explanation, it's best to experience the work in code to better understand.

9. Technology Stack

Python (Latest Version) + Python Ext. Libraries

IDE – Currently selected Visual Studio Code.

A Computer with minimum 8GB ram to run the process efficiently as Python's run time is quite lengthy.

10. Project Setup

Disclaimer: This code is not entirely my own code; I have used some parts it as a reference tool to provide data for this research. [32]

We begin with importing the relevant libraries being:

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import pandas_datareader as web
import yfinance as yf
import datetime as dt
```

Figure 2.3: Imports

Numpy: *Used to perform mathematical operations in an array*

matplotlib.pyplot: *Used for creating a window for graphs* pandas: *Used for data science and machine learning tasks*

pandas_datareader: *Used for reading data easily*

yfinance: *Used to access Yahoo Finance API*

datetime: *Used for Date-Time references*

Along with these there are some libraries that you are required to install.

1. Open the command prompt via Start > Typing "cmd" in search or opening "Run" and typing "cmd" and clicking Enter on the keyboard.
2. Next type the following: pip install x ; where instead of x we will replace with the following; hence it will look like the following:

```
C:\Users\Smith>pip install numpy matplotlib pandas pandas-datareader tensorflow scikit-learn
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: numpy in c:\users\smith\appdata\roaming\python\python311\site-packages
)
Requirement already satisfied: matplotlib in c:\users\smith\appdata\roaming\python\python311\site-pac
```

Figure 2.4: Install command executed

```
[notice] A new release of pip is available: 23.1 -> 23.1.2
[notice] To update, run: python.exe -m pip install --upgrade pip

C:\Users\Smith>python.exe -m pip install --upgrade pip
```

Figure 2.5: (Upgrade if required.)

After installing the above the IDE will recognize the commands and imports and will allow you to progress forward.

Now we load in the data. Here we are using Bitcoin's data, we can use any other company's ticker symbol to download their data from Yahoo.

```python
#load data
companyid = 'BTC-GBP'

starttime = dt.datetime(2012,1,1)
endtime = dt.datetime(2020,1,1)

#data = web.DataReader(companyid, 'yahoo', starttime, endtime)

data = yf.download(companyid, starttime , endtime)
```

Figure 2.6: Loading the Data

Only the share closing price will be used for our model prediction. Our window length for predicting the next price is the last 60 days of closing price data. Any length will work if you do your research according to your guidelines.

```python
# Preparing the data
scaler = MinMaxScaler(feature_range=(0,1))
scaled_data = scaler.fit_transform(data['Close'].values.reshape(-1,1))

# How many days of data we are evaluating for prediction
prediction_days = 60

# Creating variables for taking the values in the array
x_train = []
y_train = []

for x in range(prediction_days, len(scaled_data)):
    x_train.append(scaled_data[x - prediction_days:x, 0])
    y_train.append(scaled_data[x, 0])

x_train, y_train = np.array(x_train), np.array(y_train)
x_train = np.reshape(x_train, (x_train.shape[0], x_train.shape[1], 1))
```

Figure 2.7: Preparing the data for training

The model can be configured in any way that is most suitable for you, so you do not have to use this configuration to train and predict. Various hyperparameter values can be experimented with, including number of neurons, layers, and dropouts...

```python
# Building and Training the model

model = Sequential()

model.add(LSTM(units=50, return_sequences=True, input_shape=(x_train.shape[1], 1)))
model.add(Dropout(0.2))
model.add(LSTM(units=50, return_sequences=True))
model.add(Dropout(0.2))
model.add(LSTM(units=50))
model.add(Dropout(0.2))
#predition of the next stock price

model.add(Dense(units=1))

model.compile(optimizer='adam', loss='mean_squared_error')
model.fit(x_train, y_train, epochs=25, batch_size=32)
```

Figure 2.8: Making the model to predict using the existing data

```python
''' Test the model accurency on existing data'''
test_starttime = dt.datetime(2020,1,1)
test_endtime = dt.datetime.now()

test_data = yf.download(companyid, starttime , endtime)
actual_prices = test_data['Close'].values

total_dataset = pd.concat((data['Close'], test_data['Close']))

model_inputs = total_dataset[len(total_dataset) - len(test_data) - prediction_days:].values
model_inputs = model_inputs.reshape(-1,1)
model_inputs = scaler.transform(model_inputs)
```

Figure 2.9: To see how well the model has learned, we execute the program to predict the data.

```python
# Make predictions on test data

x_test = []

for x in range(prediction_days, len(model_inputs)):
    x_test.append(model_inputs[x - prediction_days:x,0])

x_test = np.array(x_test)
x_test = np.reshape(x_test, (x_test.shape[0], x_test.shape[1], 1))

predicted_prices = model.predict(x_test)

predicted_prices = scaler.inverse_transform(predicted_prices)
```

Figure 3.0: Making prediction on test data

```python
# Plot the test predictions
plt.plot(actual_prices, color = 'black', label=f"Actual {companyid} Price")
plt.plot(predicted_prices, color = 'green', label=f"Predicted {companyid} price")
plt.title(f"{companyid} share price")
plt.xlabel('time')
plt.ylabel(f"{companyid} share price")
plt.legend()
plt.show()
```

Figure 3.1: Visualizing the predictions.

As we can see here with this simple configuration and only with 100 epochs our model is able to follow the actual pattern, but it consistently results in a lower trend value than the actual price except in a few instances when it almost overlaps with the mean predicted value, which is very close to the actual price.

```python
# Predict next day

real_data = [model_inputs[len(model_inputs) + 1 - prediction_days:len(model_inputs)]]
real_data = np.array(real_data)
real_data = np.reshape(real_data, (real_data.shape[0], real_data.shape[1],1))

prediction = model.predict(real_data)
prediction = scaler.inverse_transform(prediction)
print(f"The prediction value for the next day is : {prediction}")
```

Figure 3.2: Predicting the Next Day price.

Finally, we run the application to which we receive this information.

```
[*********************100%***********************]  1 of 1 completed
Epoch 1/25
59/59 [==============================] - 18s 92ms/step - loss: 0.0080
Epoch 2/25
59/59 [==============================] - 6s 98ms/step - loss: 0.0028
Epoch 3/25
16/59 [=======>......................] - ETA: 3s - loss: 0.0025
```

Figure 3.3: Program process while execution.

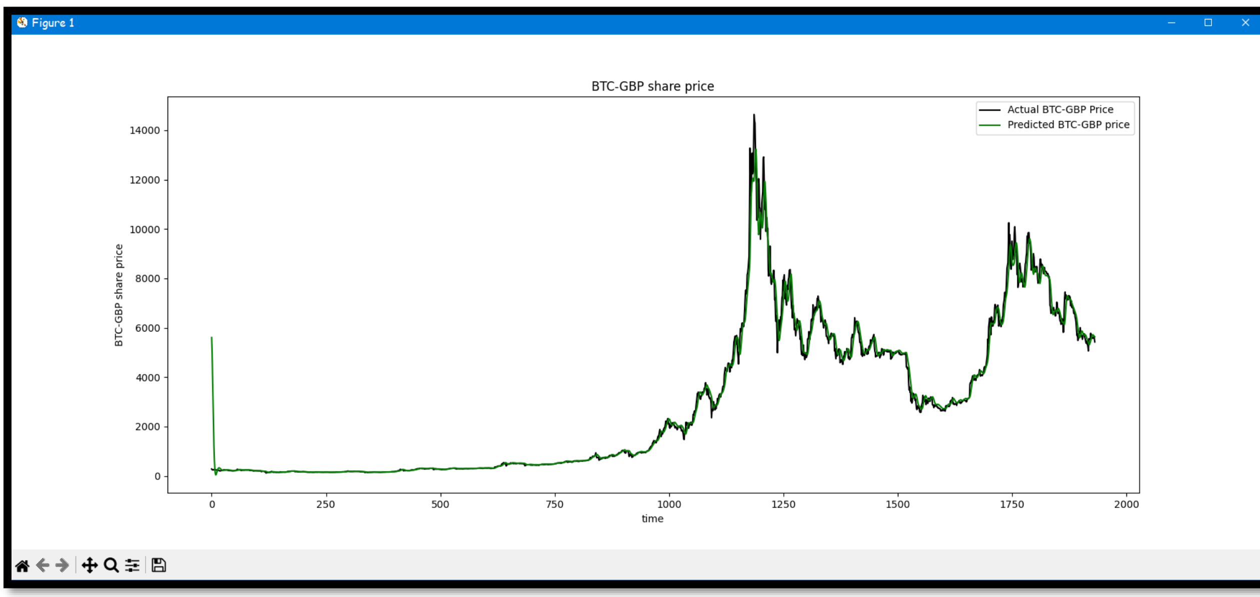

Figure 3.4: Prediction graph produced.

Upon further inspection, we can conclude that the current Bitcoin's share price is: $29,330.76 USD on 27th April 2023.

The prediction value for the next day is: [[13792.01]]

I will be trying two methods, one the normal method where it analyses the 60 days as normal, and the latter being reiterating each prediction on each day, ie. it will predict the 60th day then use that information to predict the 59th day and so on.

10.1 Testing

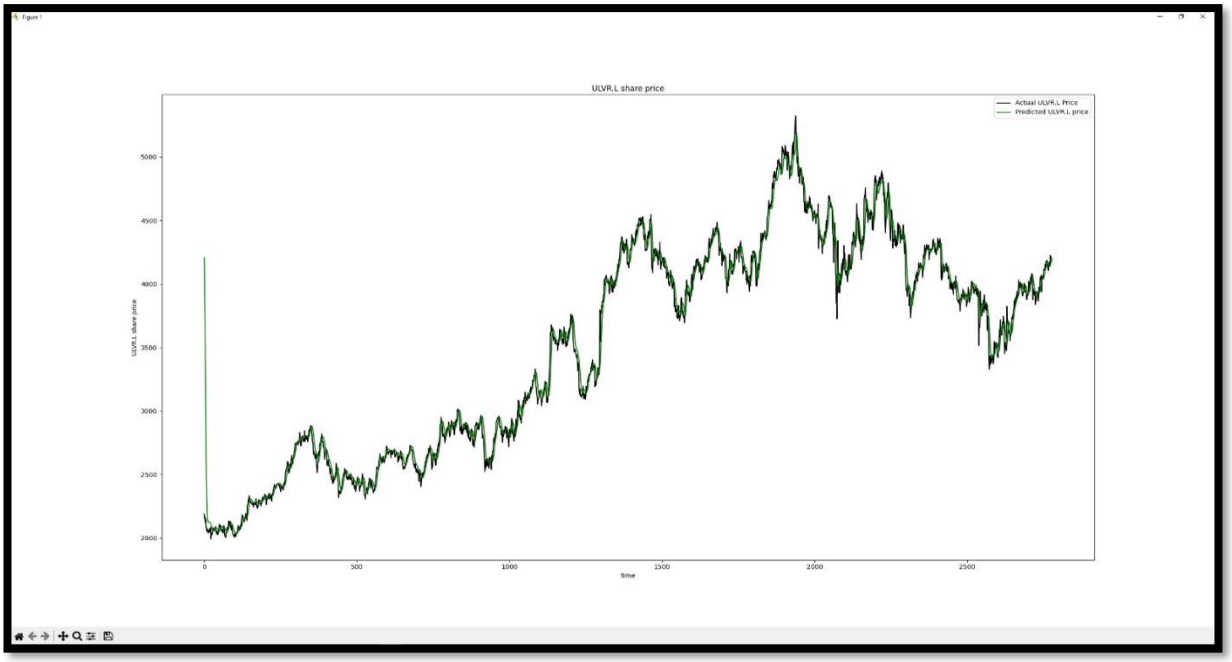

Figure 3.4.1: The above is a depiction of the ULVR.L's stock position.

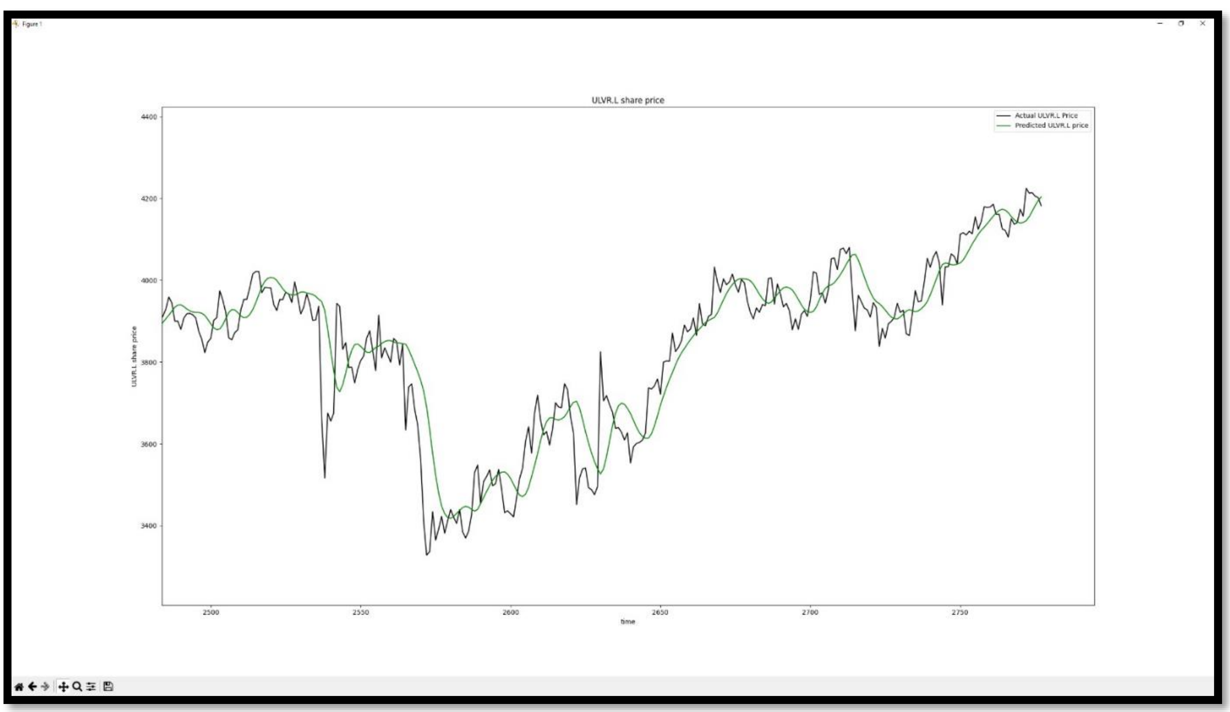

Figure 3.4.2: Current price for the stock is 4446.40 on 28th April 2023

The predicted value for ULVR.L's stock is: [4208.0825]. Difference: [238.3175]

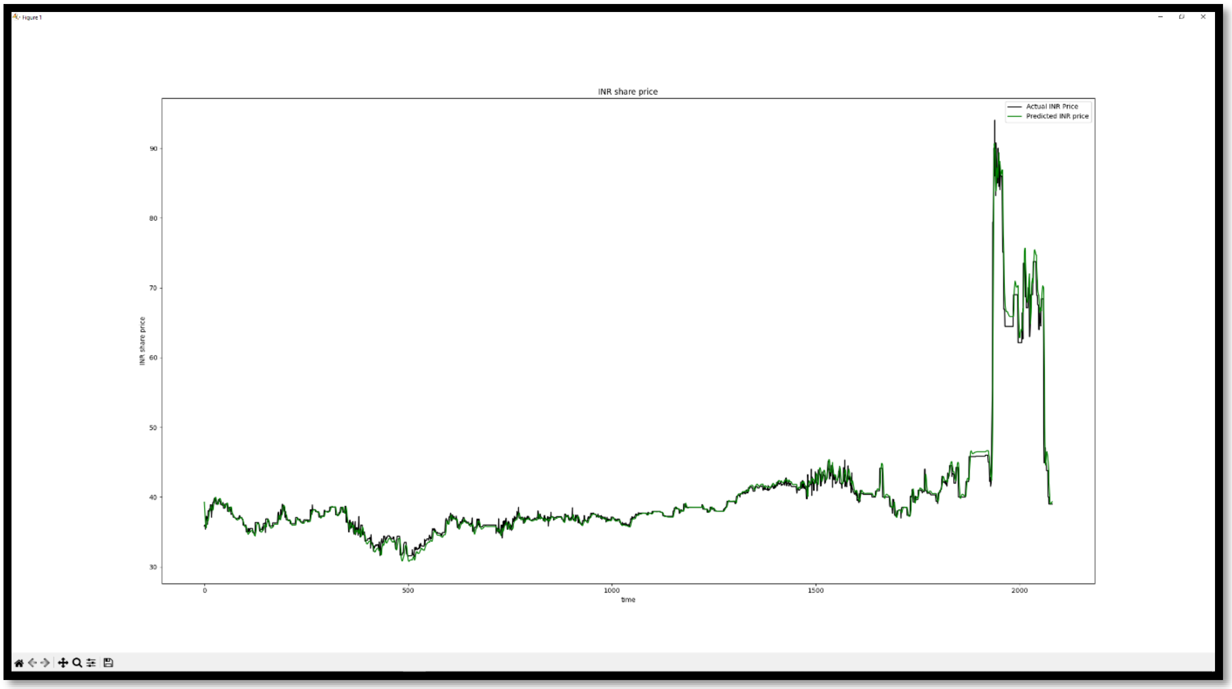

Figure 3.4.3: The above is a depiction of the INR’s stock position.

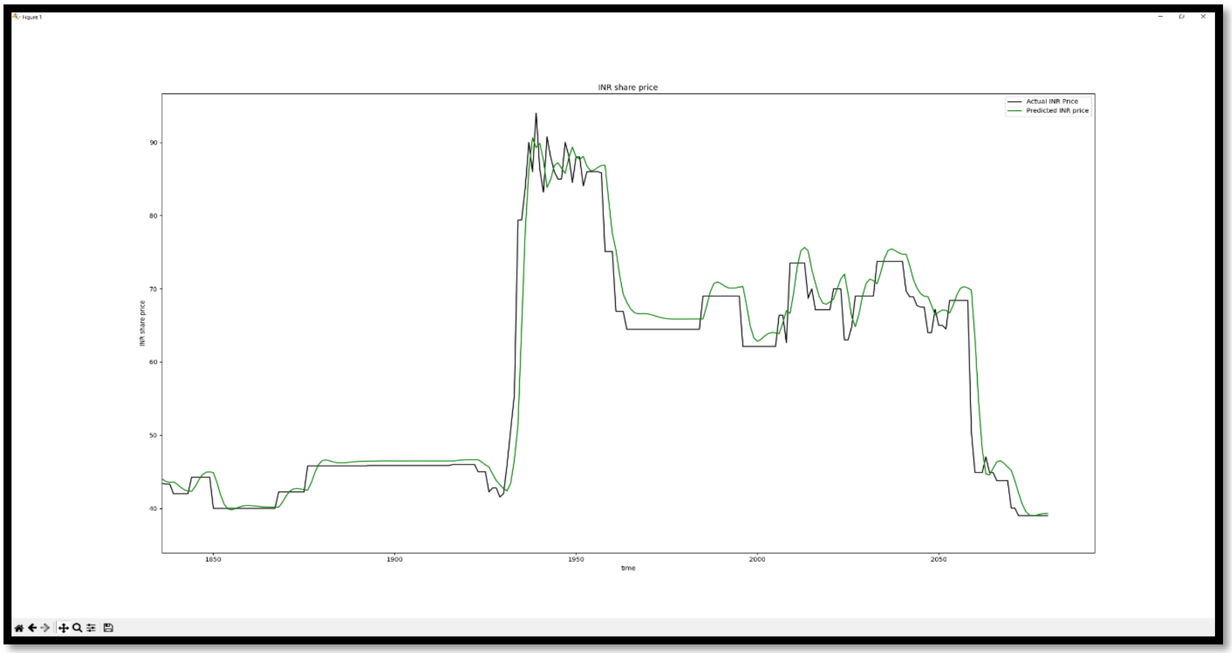

Figure 3.4.4: Current price for the stock is 39.00 on 28th April 2023

The predicted value for INR’s stock is: [39.293167]

Difference: [-0.293167]

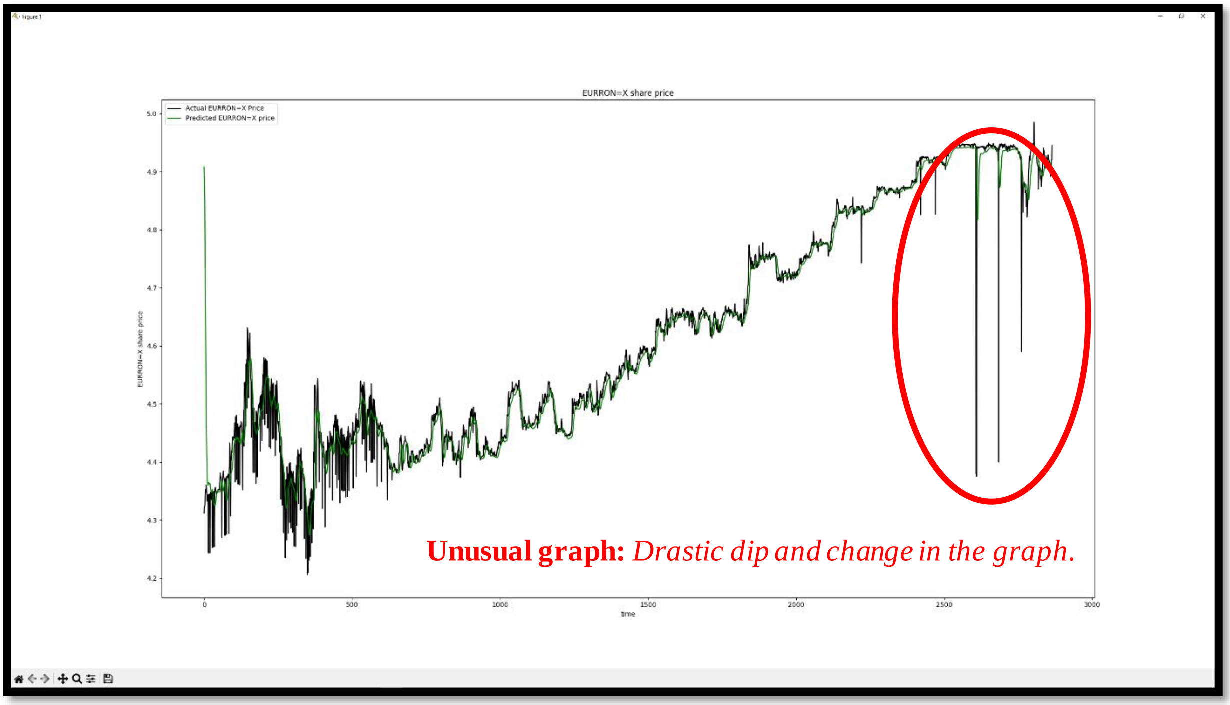

Figure 3.4.5: The above is a depiction of the EURRON=X's stock position

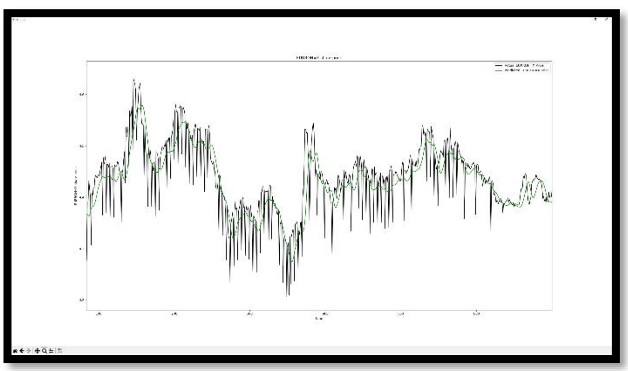

Figure 3.4.6: Current price for the stock is 4.9233 on 28th April 2023

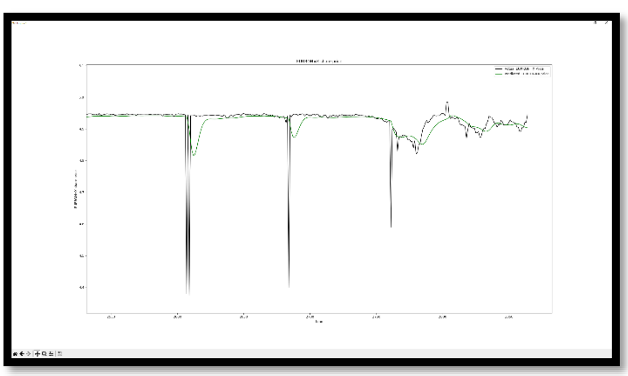

Figure 3.4.7: This indicates the sharp dips in history.

The predicted value for EURRON=X's stock is: [4.9077344]

Difference: [0.0156]

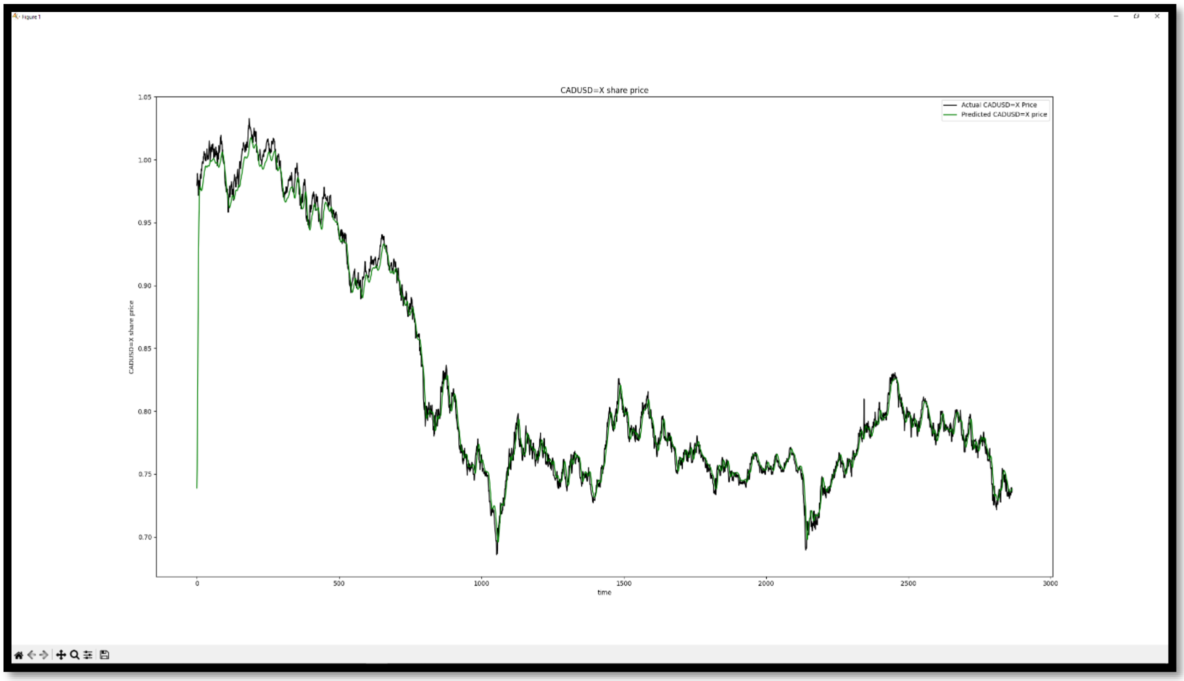

Figure 3.4.8: The above is a depiction of the CADUSD=X's stock position

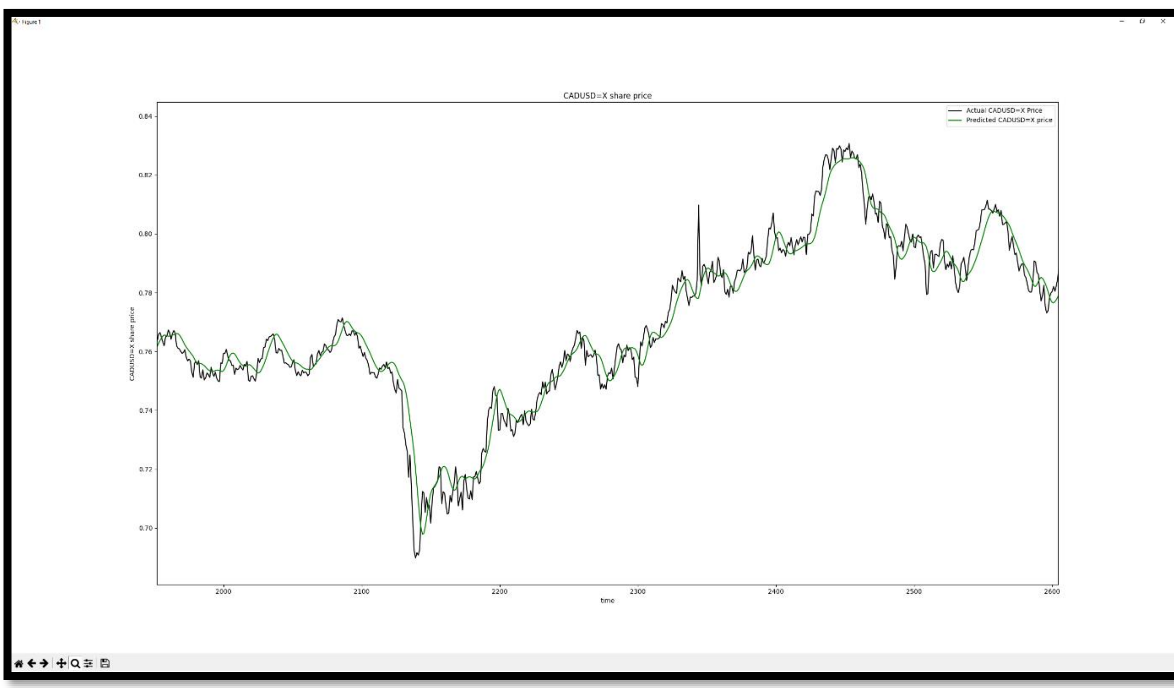

Figure 3.4.9: Current price for the stock is 0.7317 - 0.7390 on 28th April 2023

The predicted value for CADUSD=X's stock is: [0.7388896]

Difference: [-0.003]

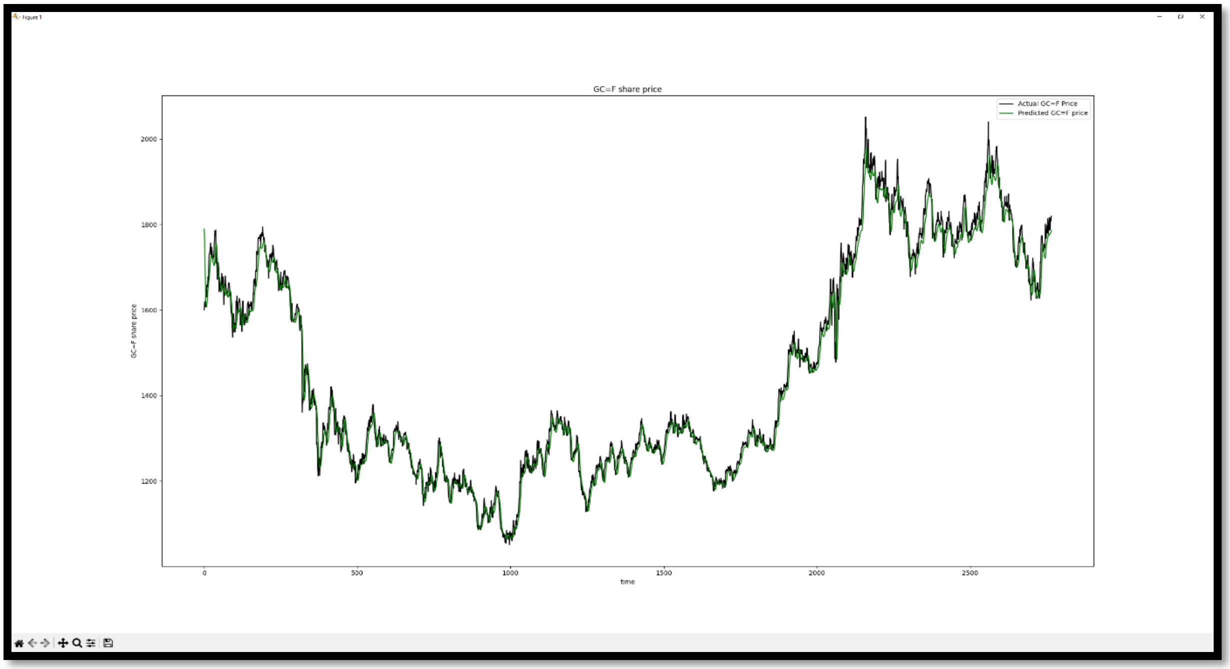

Figure 3.4.10: The above is a depiction of the GC=F’s stock position

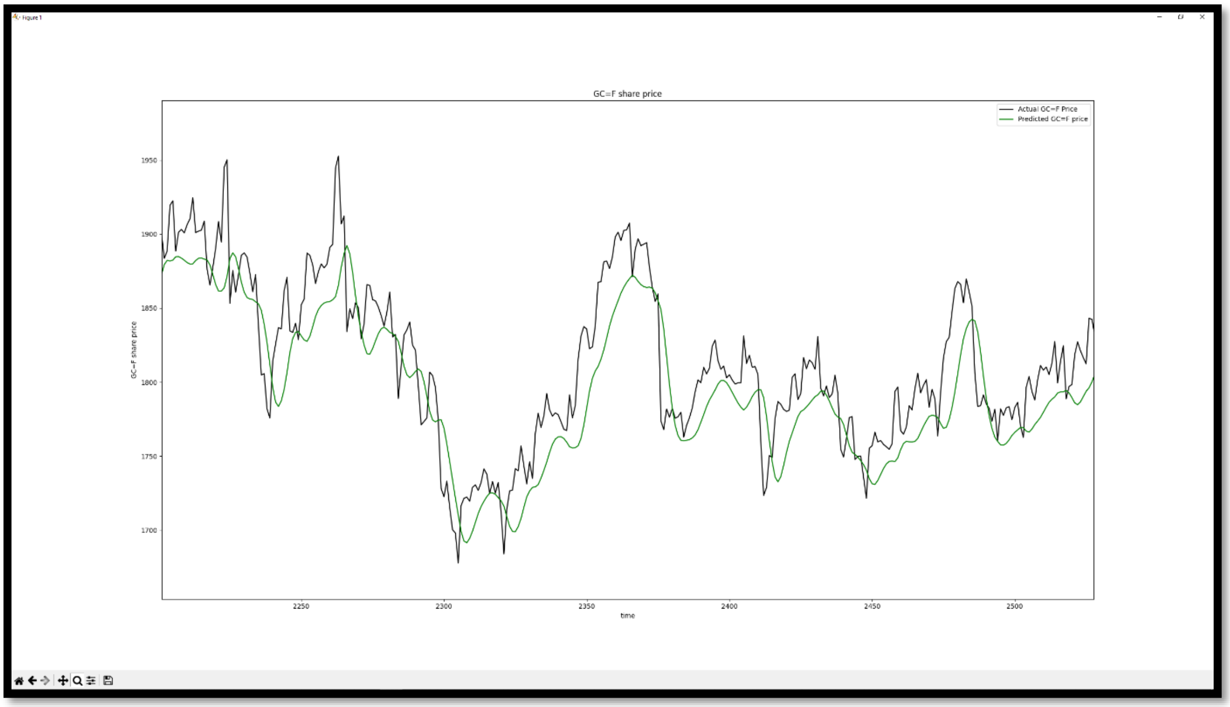

Figure 3.4.11: Current price for the stock is 1.17 THB on 28th April 2023

The predicted value for GC=F’s stock is: [1789.3992] Difference: [-0.619]

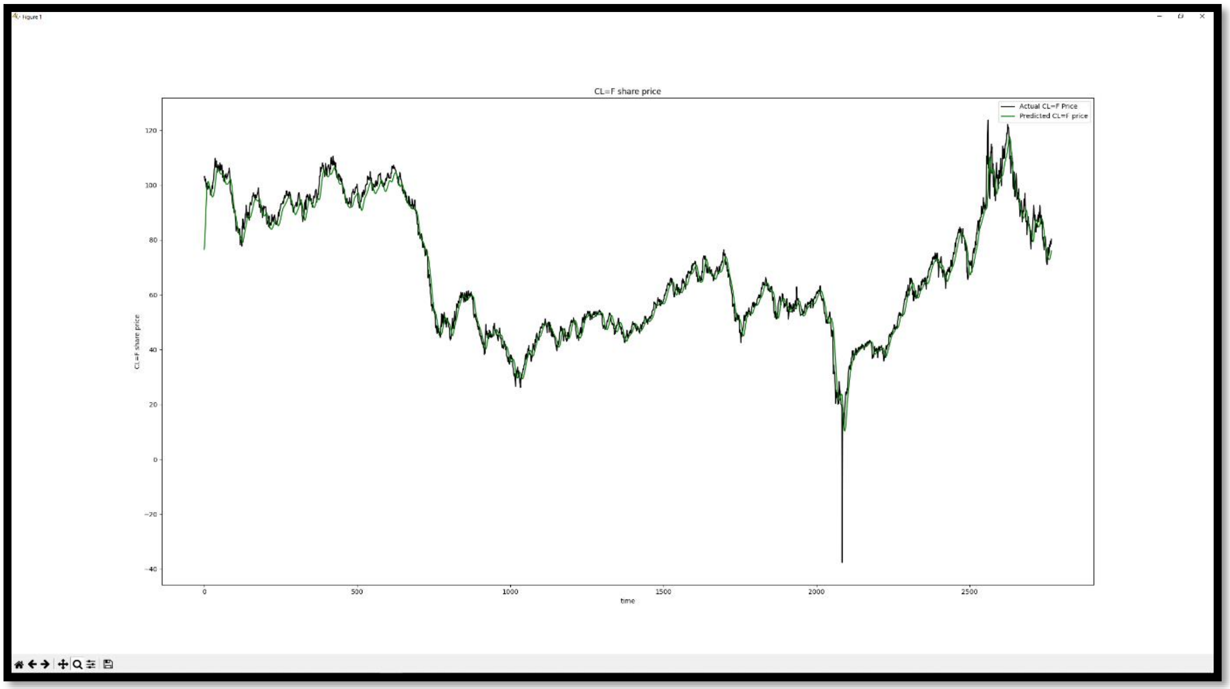

Figure 3.4.12: The above is a depiction of the CL=F's stock position

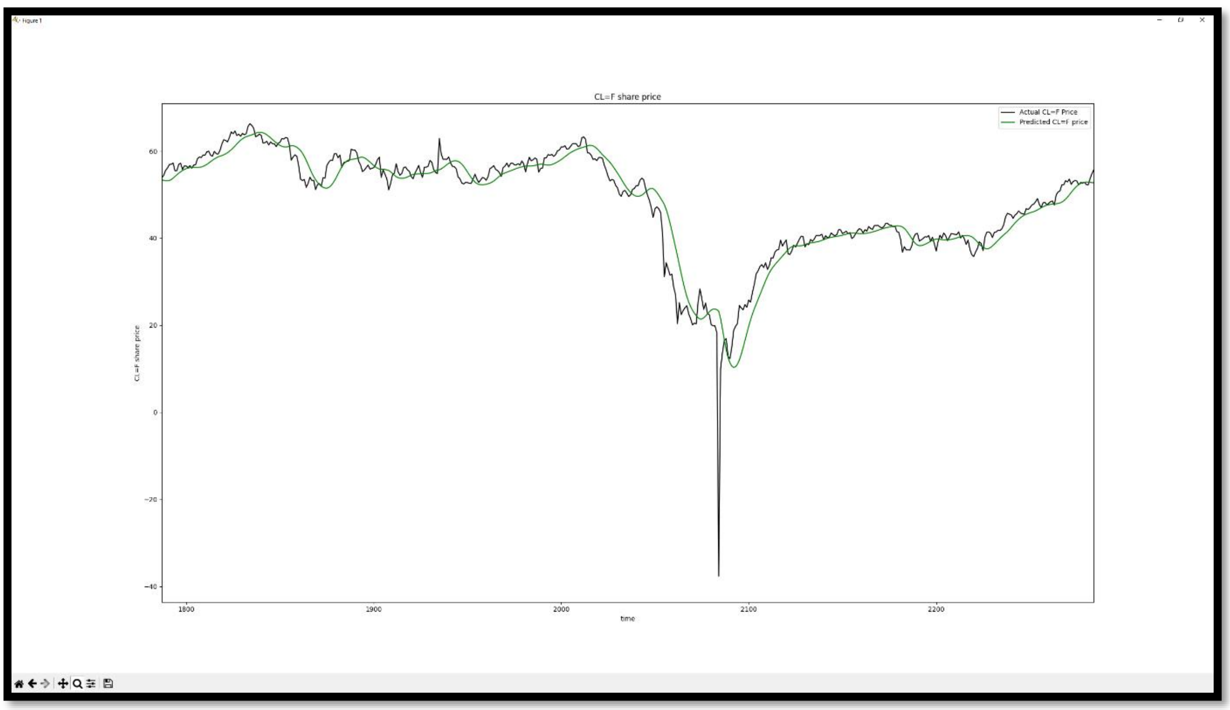

Figure 3.4.13: Current price for the stock is 74.76 on 28th April 2023

The predicted value for CL=F's stock is: [76.54916] Difference: [-1.78916]

We have a difference between the actual and predicted value. We can always do more experimentation to reduce that gap and make models more efficient.

Test #2 Reiteration: After repeating the process, the results were highly inaccurate, what seemed good at the time, hindsight would have saved me the time that I would put in this as it seems like a waste. It was a waste of time due to the very fact that I realized late that it doesn't have data to begin with, and if we start the prediction process this early, the prediction process will increase its level of randomness as there is insufficient data. Nevertheless, I have recorded it so no one else needs to do the test.

Another issue I ran into was that the process is very long, time-consuming, and not very efficient.

Test #2 conclusion: After attempting such a process. The result was highly inaccurate as it did not have enough clear base data to pinpoint the irregularities.

10.1.1 ARIMA Modelling [43]

ARIMA (p, d, q): The ARIMA model is the largest class of models for predictive time series and can be stabilized by variables such as variance and intersection. The full form of ARIMA is "Autoregressive Integrated Moving Average". The lagged term of the variance seen in the estimation equation is called the "autoregressive" term, the lagged term of the estimation error is called the "moving average" time and the time series that must be different to remain stationary. The "synthetic" version of the fixed series. Arbitrary walk and stochastic models, autoregressive models, and exponential smoothing models (For example: Exponentially Weighted Moving Average), are special cases of the ARIMA model. A non-seasonal ARIMA model is classified as an "ARIMA (p,d,q)" model, where:

- p is the number of auto-regressive terms,
- d is the number of non-seasonal differences,
- q is the number of lagged forecast errors in the prediction equation.

ARIMA (p, d, q) or can be expressed as:

$$\mathrm{Xt} = \theta 0 + \varphi 1\mathrm{Xt} - 1 + \varphi 2\mathrm{Xt} - 2 + \varphi \mathrm{pXt} - \mathrm{p} + \mathrm{et} - \theta 1\mathrm{et} - 1 - \theta 2\mathrm{et} - 2 - \theta \mathrm{qet} - \mathrm{q}\ ,\ldots(1)$$

where Xt and et are the actual value and random error at time t, respectively; φi(i=l, 2 , ... , p) and ϑj(j=1, 2 , ... , q) are model parameters. p and q are integers and often referred to as orders of autoregressive and moving average polynomials.

10.1.2 ARIMA based Recurrent Networks Forecasting System [44]

A general linear model used for forecasting purposes is the class of ARIMA(p,q)

$$x_t = r + \sum_{i=1}^{p} \varphi_i x_{t-i} \sum_{j=1}^{q} \theta_j e_{t-j} + e_t \qquad \text{…………………(1)}$$

Figure x : This equation is referenced as Eq. 1 for the calculation below

where it is assumed that E ($e_t | x_{t-1}, x_{t-2}, \ldots$) = 0, [44] This condition is satisfied when e_t, are zero mean, independent and identically distributed, and are independent of past x_t. In this case, the minimum mean squared error predictor is $\hat{x}_t$ = E ($x_t | x_{t-1}, x_{t-2}, \ldots x_1$) optimal ARIMA predictors is given by

$$\hat{x}_t = \varphi_1 x_{t-1} + \ldots.. + \varphi_p x_{t-p} + \theta_1 \hat{e}_{t-1} + \ldots.. + \theta_q \hat{e}_{t-q}, \text{ where}$$
$$\hat{e}_{t-j} = x_{t-j} - \hat{x}_{t-j} \qquad j = 1,2,\ldots,q. \qquad \text{………………. (2)}$$

Figure y: This equation is referenced as Eq. 2 for the calculation below

We are developing an ARIMA-based recurrent network for prediction purposes; the structure of the network is shown in equation 2, Figure y. The basic architecture originates from [1], but the feature data. used and the way of error feedback are different. The number of the input nodes of the recurrent network equals $p + q$, and number of hidden nodes is selected by trial and error. [44] The output of the network approximates conditional mean predictor and is given by:

$$\hat{x}_t = \sum_{i=1}^{h} W_i f\left[\sum_{j=1}^{p} w_{ij} x_{t-j} + \sum_{j=1}^{q} w'_{ij}\left(\hat{x}_{t-j} - \hat{x}_{t-j-1}\right) + \theta_i\right] \qquad \text{…..……………(3)}$$

Figure z : This equation is referenced as Eq. 3 for the calculation below and here f is the sigmoidal function.

Note that in equation 3, Figure z, the input training data (after second difference) are stationary. Instead of using x_t - $\hat{x}_t$ as in [1], we use the feedback equation êt - $\hat{x}_t$ - $\hat{x}_{t-1}$ as the next new input data for a unit-delay node. The reason we do this is that any two successive data must have the largest correlation, according to ACF plots shown in equation 1, Figure x.

We will use $\hat{e}_t$ - $\hat{x}_t$ - $\hat{x}_{t-1}$ not only during back propagation training, but also during the recall (i.e., testing) process. We believe that the network could suffer a decrease in prediction accuracy without referring to any error data $\hat{e}_t$ during the testing phase. Furthermore, there is no way of knowing the value of x_t in advance, this eliminates possibility of using x_t - $\hat{x}_t$ during the testing phase.[44]

10.2 ANN

According to the results short-term trend predictions made by this method have been very precise. Unfortunately, RFE is unable to detect nor recognize, and eliminate redundant features hence it is out as an algorithm. Artificial Neural Network (ANN) models can theoretically mimic human brain activity. A review article by Ritika Chopra and Gagan Deep Sharma in 2021 compiled results from over 150 research studies related to ANN. [23]

"The authors have classified the models into seven themes: stock market covered; input data; data pre-processing; artificial intelligence technique; training algorithm; performance measure; and nature of the study. We observe that neural and hybrid-neuro approaches are ideal for stock market forecasting since they provide more accurate outcomes than the traditional models used in most studies." [23] -by Ritika Chopra and Gagan Deep Sharma (2021)

ANN has displayed the most efficient results through the tests. The way ANN works is the model works with three layers. It consists of an input layer, hidden layer, and a output layer. The input layer consists of variables such as H-L, O-C and 7 DAYS MA, 14 DAYS MA, 21 DAYS MA, 7 DAYS STD DEV and Volume [23].

*Keywords: **H-L**: Stock High minus Low Price, **O-C**: Close minus Open Price, **XDAYS MA**: Stock Price's X Days Moving Average (X is the number of days), **7DAYS STD DEV** : Stock Price's Standard Deviation for the Past Seven Days.[23]*

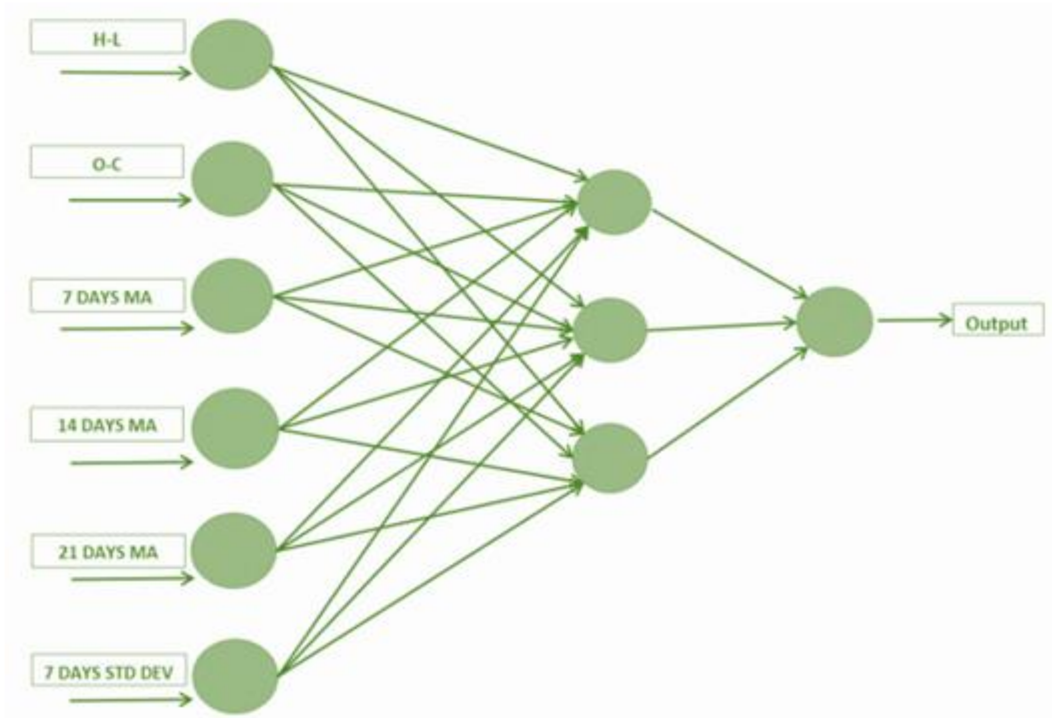

Figure 3.5: Prediction of stock prices depicted using Artificial Neural Network (ANN)

From the figure it's understandable that all the data is taken, then the noise is removed, to further get precise information to which it is then further broken down to even simpler information which is then processed to the output. This method was the most efficient when comparing results with other algorithms such as RF (Random Forest).

We can receive values by calculating the RMSE (Root Mean Square Deviation), MAPE (Mean absolute percentage error) and MBE (Mean Bias Error).

RMSE is computed using equation 1.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(O_i - F_i)^2}{n}} \qquad \ldots\ldots\ldots(1)$$

where, where 'Oi' refers to the original closing price, 'Fi' refers to the predicted closing price and 'n' refers to the total window size.

MAPE has also been used to evaluate the performance of the model and is computed using equation 2.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{(O_i - F_i)}{O_i} * 100$$
………(2)

where, where ‘Oi’ refers to the original closing price, ‘Fi’ refers to the predicted closing price and ‘n’ refers to the total window size.

MBE has also been used to evaluate the performance of the model and is computed using equation 3.

$$MBE = \frac{1}{n}\sum_{i=1}^{n}(O_i - F_i)$$
………(3)

Where, where ‘Oi’ refers to the original closing price, ‘Fi’ refers to the predicted closing price and ‘n’ refers to the total window size.

Figure 3.6 represents graphs showing original closing price of stock with respect to predicted closing price of stock of three different companies using ANN.

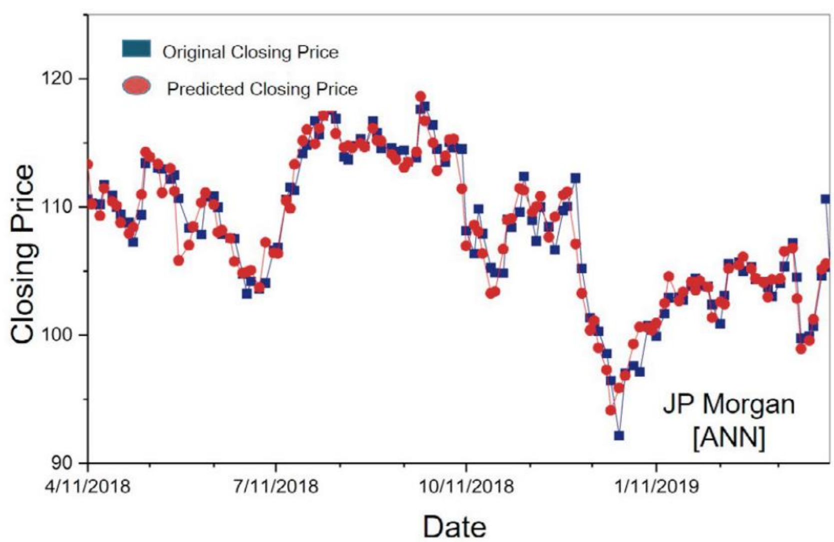

Figure 3.6: ANN graph for JP Morgan

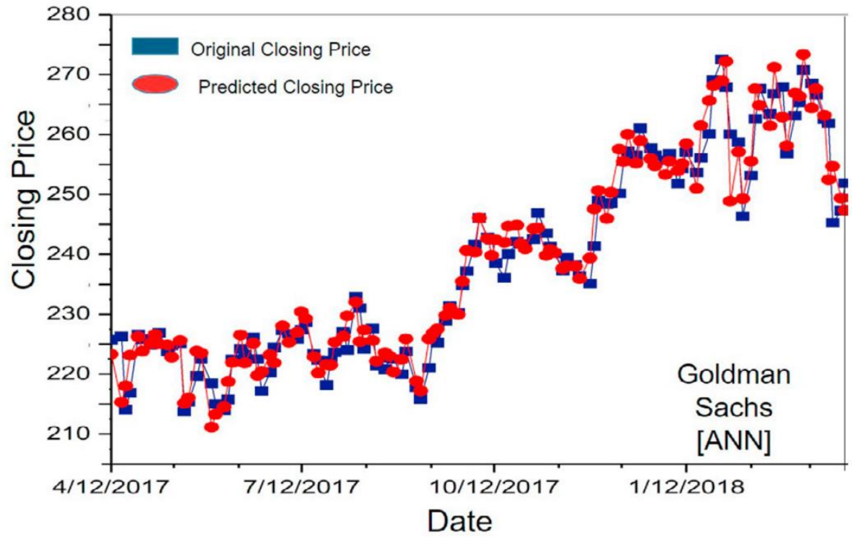

Figure 3.7: ANN graph for Goldman Sachs

Figure 3.8: ANN graph for Nike

Table		ANN	
Company	RMSE	MAPE	MBE
Nike	1.10	1.07%	-0.0522
Goldman Sachs	3.30	1.09%	0.0762
JP Morgan and Co.	1.28	0.89%	-0.0310

According to the comparative analysis, ANN provides better RMSE and MAPE values for Nike, Goldman Sachs and JP Morgan and Co.

As a conclusion, ANN provides the best results according to *Mehar Vijh et al. / Procedia Computer Science 167 (2020) 599–606*.

10.3 GAN

Another algorithm would include **GAN** (Generative Adversarial Networks) however it was concluded as "*Despite having obtained promising results and a stable architecture, training GANs remains a difficult task due to the need of tuning many hyperparameters while keeping a balance between the generator and the discriminator network.*" [18]

The main section of the research done regarding GAN was attempting to make accurate forecasts on the closing price, however, their efforts were less accurate if not closer to what LTSM could optimally provide. [18]

The value of a stock is often seen as a time series model and therefore time series analysis [17] is one of the most popular models for forecasting stock prices.

Analysis of Major Contributions [33]

The numerous methods applied for achieving share price prediction are broadly divided into four categories:

- **Traditional Machine Learning Methods -** Includes traditional methods such as linear regression analysis and logistic regression analysis.
- **Deep Learning and Neural Networks -** Many of these techniques make use of RNNs and LSTMs which are a special type of RNN.
- **Time Series Analysis Methods** - This method depends on forecasts and projection of discrete time data.
- **Graph-Based Approaches -** Often concerned with comparing the stock market as a network of interconnected nodes where a change in one component will impact the prices of other components.

Table 1. Summary of various algorithms and their performance for predicting stock prices [33]

Category	**Algorithm**	**Metrics Used**	
Traditional Machine Learning	Partial Least Squares Classifier (PLS Classifier) [15]	Average Error Value = 0.81225	
	Sequential Minimal Optimization (SMO) [15]	Average Error Value = 0.79656	
	Sentiment Analysis of Tweets using SVM and Extremely Randomized trees (ExtRa) [23]	R2 similarity and RMSE of the Ensemble model with respect to the SVM regressor. The model best performed for a 5-month dataset with difference in R^2 values given as:	
		SVM Meta-Regressor: +0.003	ExtRa tree Meta Regressor: -0.02
Deep Learning	LSTM coupled with sentiment analysis [22]	Matthews Correlation Coefficient (MCC) = 0.04092	Accuracy = 52.27%
	Attention-based LSTM with sentiment analysis [22]	Matthews Correlation Coefficient (MCC) = 0.04780	Accuracy = 54.58%
	LSTM [24]	RMSE = 0.0091, MAE=63 , MAPE=2.23%	
	CNN [24]	RMSE = 0.0087, MAE=61, MAPE=2.16%	
	Conv1D-LSTM [24]	RMSE = 0.0081, MAE= 56, MAPE=1.98%	
Time Series	ARIMA[20]	For 3 time steps ahead: RMSE: ~15% MAPE: 20-25% MAE: ≤15%	For 9 time steps ahead: RMSE: 15-20% MAPE: 15-20% MAE: 10-15%
	Generalized Additive Model (GAM) using Fourier transformations to identity seasonality of trends combined with MLR [19]	Accuracy ~ 70% MAPE = 1-5%	
	Time series supervised learning [25]	K-fold cross validation accuracy for: Perceptron NN = 75.48% SVM = 75.48% Logistic Regression = 89.45%	Train-Test split accuracy for: Perceptron NN = 76.68% SVM = 89.33% Logistic Regression = 89.33%

		Type I Error for: Perceptron NN = 15.65% SVM = 10.66% Logistic Regression = 10.01%	Type II Error for: Perceptron NN = 17.83% SVM = 10.68% Logistic Regression = 10.14%
Graph-Based	Hierarchical Graph Attention Network [21]	Average Accuracy = 0.3948	Average F1 Score = 0.3389
	Ensemble of Graph Theory [20]	For 3-time steps ahead: GCN with Pearson Correlation: • RMSE: 10-15% • MAPE: ~15% • MAE: 5-10% GCN with Spearman Rank Correlation: • RMSE: 10-15% • MAPE: ~5% • MAE: 5-10% GCN with Kendall Correlation: • RMSE: ~10% • MAPE: <5% • MAE: ≤10%	For 9-time steps ahead: GCN with Pearson Correlation: • RMSE: 10-20% • MAPE: 10-20% • MAE: ~10% GCN with Spearman Rank Correlation: • RMSE: 10-20% • MAPE: <10% • MAE: <10% GCN with Kendall Correlation: • RMSE: 10-20% • MAPE: <10% • MAE: <10%

Table 2. Merits and demerits of the different approaches used for forecasting stock prices. [33]

Category	Merits	Demerits
Traditional Machine Learning Algorithm	Traditional ML algorithms, particularly SVM, yield relatively higher accuracy as they work well with datasets of high dimensionality.	These algorithms exhibit high sensitivity to outliers.
Deep-Learning	RNNs and LSTMs are the go-to Deep Learning algorithms employed for the task. RNNs offer an advantage as they capture the context of the data while training. LSTMs perform reasonably well as they can correlate the non-linear time series data in the delay state. [26]	Requires high training time and large memory requirements.
Time-Series	Time-series forecasting techniques such as ARIMA work well for linear data and provide reliable stock price forecasts for the short term. [5]	The model may not yield dependable results for long term forecasts of stock prices.
Graph-Based	Focuses on forming relationships based on correlation and causation among the nodes which is useful for exploring previously hidden insights and aids informative decision making.	Traditional ML algorithms still outperform graph based approaches in terms of accuracy.

Initially I ought to create a program of my own aspects would be another attempt in multiple publications however realizing certain aspect on don't fix what isn't broken and focus on not to re-invent the wheel; I began on focusing how else to better improve the analysis and accuracy and impact.

Now that we know how an AI can make a prediction; My next objective was to measure the margin of error, what caused the gap in error. Which parameters caused it and if we can manage or control it to manipulate the outcome in a positive way.

When mentioning the parameters, it can be confused with the variables in the code; I am not referring to them, I am talking about how the news and natural environmental changes affect the stock market.

What I found in the research was different methods and formulas on how to narrow down the accuracy between the predicted values and actual values.

Going through multiple research papers I realized only a couple of them showed major emphasis on the parameters, they did mention it however there was limited data recorded on the relative impact it had on the market.

Short-term movements in stock prices are highly influenced by news. According to a detailed analysis, dissemination through the media can play an essential role in influencing stock prices, leading to emotional investors easily affected by the news. The news revealing a scandal at a company, for example, can significantly affect investors' attitudes, resulting in the price of the stock declining as a result, it is extremely difficult for the investors to sell the stocks.

Due to the herd effect theory, investors are more likely to take similar investment actions as others when facing a new event, making the change of stock price easier to track. Every day, stock market news is updated, and a wide variety of news is exposed to investors. An emotional reaction is likely to occur because of this news. When making investment decisions, investors are susceptible to market emotions. Considering this, the topic of analyzing the impact of news on stock price has become trending in research.

When evaluating the stock market's reaction to new information, two factors are important to note: the reaction time and the extent of the reaction.

Some real time examples of impacts we can consider are:

10.4 Russia

Saudi Price War: Prior to opening on 9 March 2020 (Monday), the Dow Jones Industrial Average futures market fell over 1,300 points and suspended trading as a result due to a combination of coronavirus concerns and the oil price war. [20]

Covid-19: In 2020, the price of Urals experienced a huge decrease at the beginning of the year due to the coronavirus (COVID-19) pandemic, dropping to as low as 16.6 U.S. dollars per barrel in April.

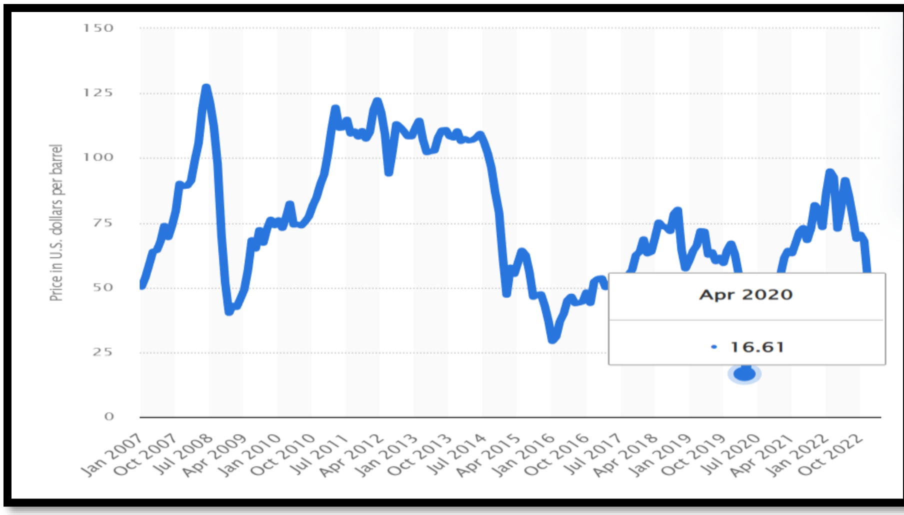

Figure 3.9: Lowest cost recorded.

Macroeconomic concerns such as energy-price-led inflation and rising rates, coupled with the political uncertainty caused by the war, led to equity-market drawdowns that lasted as late as September 2022 for all major global markets. [46]

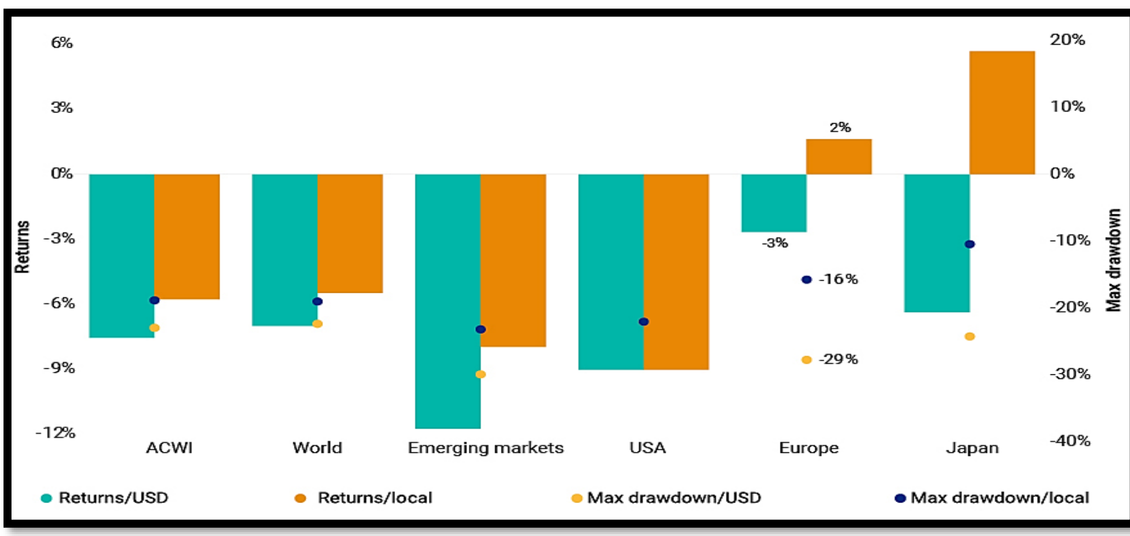

Figure 4.0: Equities markets in Europe and globally over the past year

Invasion by Russia appears to have marked the end of that expansion and the beginning of a new era of higher rates and higher returns. In the past few years, a substantial gap had encouraged capital into the sector, but the rapid rise in swap rates forced debt costs to surpass property yields.

Europe was most affected by rising interest rates, particularly Germany because of its reliance on Russian energy. In April, German property sales were at a six-year low, and they continued to drop throughout 2022. Despite this, global volumes of transactions were down by 64% in the fourth quarter, and only a few countries were spared by a substantial drop in liquidity due to the divergence between buyer and seller price expectations. [46]

The continued easing of inflation and stabilization of bond yields and debt costs will provide more certainty in the short term and give buyers and sellers more confidence in their buy-sell-hold decisions.[46]

Real estate is now part of a redesigned investment landscape once this temporary disruption is over. When interest rates and bond yields were low, core property was effectively a bond substitute; however, they will probably rise again, as they did last cycle. In a multi-asset context, property may serve a different function.

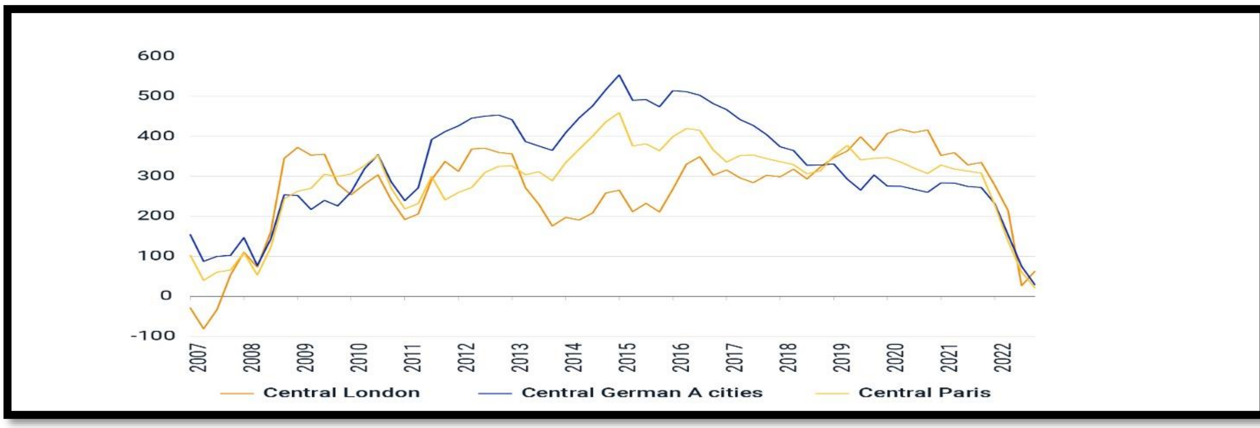

Figure 4.1: Property's competitive advantage has been eroded by rising rates.

10.5 COVID 19

There were more than 1,100 daily stock market moves (up or down) greater than 2.5 percent from 1900 to 2019. Next-day newspaper accounts attribute not even one of these jumps to infectious disease outbreaks or pandemic related developments. From 24 February to 20 April 2020, newspapers attribute two dozen such jumps to coronavirus-related developments. A similar pattern holds for measures of real economic activity. [21]

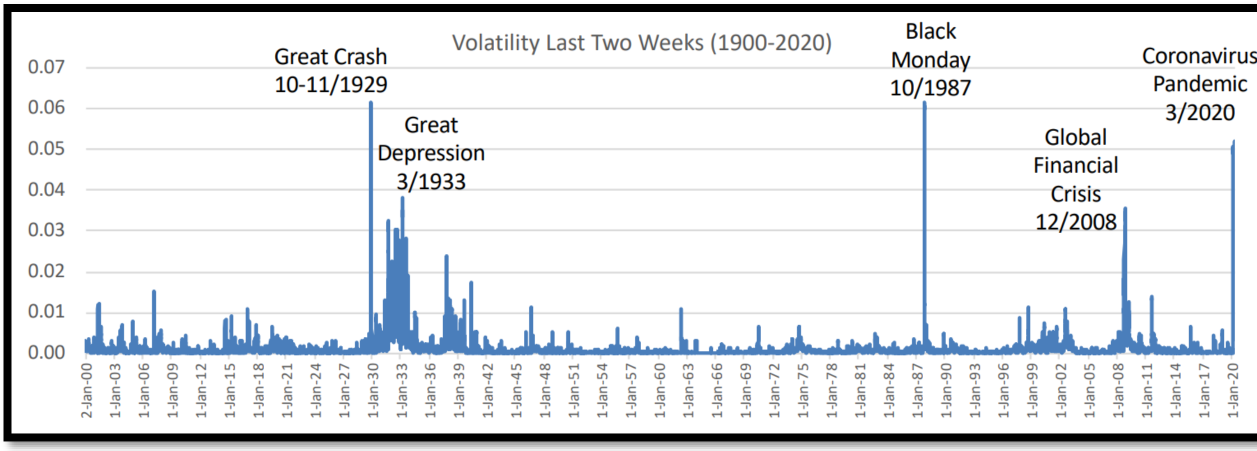

Figure 4.2: Volatility of the U.S. stock market between 1900 & 2020 on realized prices.

	Number of Daily U.S. Stock Market Jumps Greater than |2.5%|	Number Attributed to Economic Fallout of Pandemics	Number Attributed to Policy Responses to Pandemics
2 January 1900 to 21 February 2020	1,116	0	0
24 February 2020 to 30 April 2020	27	13.4	10.4

Figure 4.3: Values of stock market jumps from 2nd Jan 1900 to 30th April 2020

Considering that the Covid-19 cases and deaths have increased day by day in line with the results obtained, it is understood that investing in the stock market is not the right option for investors. Investing in gold markets, which is seen as a safe haven in all financial markets, can be seen as a logical choice for investors. [21]

In the above sections we can understand the decline or how the market is affected negatively from the news, being receiving critical information that might make the investor evade the option to invest.

"Due to this it also has caused a shockwave effect, affecting nearby locations. Even the cost in Zoopla and Rightmove has displayed a considerable amount of change."

\- *Anon 2023*

When referencing when the Uranium missile was given to Ukraine, a change was measured.

If we exclude the news that can influence the performance of a stock, the shift in prices is in large part affected by the conclusions that these algorithms draw from price fluctuations. [18]

10.6 Blackrock

Speaking of investors, BlackRock is one of the world's leading providers of investment, advisory and risk management solutions. They are a fiduciary to their clients. They're investing for the future on behalf of their clients, inspiring their employees, and supporting their local communities.[24]

We are focusing on Blackrock as their company was one of the first people to create a prediction-based system to predict the stock market, which got successful and hence started spreading across all divisions, now providing.

Performance Metric	3-Year Annualized Return	3-Year Cumulative Return	Rank Compared to other Funds
Top 20 Holdings Weighted	⬆ 9.05%	⬆ 29.68%	★★★★½
Top 20 Holdings Unweighted	⬆ 6.98%	⬆ 22.44%	★★★★☆
Top 50 Holdings Weighted	⬆ 7.6%	⬆ 24.58%	★★★★½
Top 50 Holdings Unweighted	⬆ 6.21%	⬆ 19.81%	★★★★☆

Figure 4.4: BlackRock performance history.

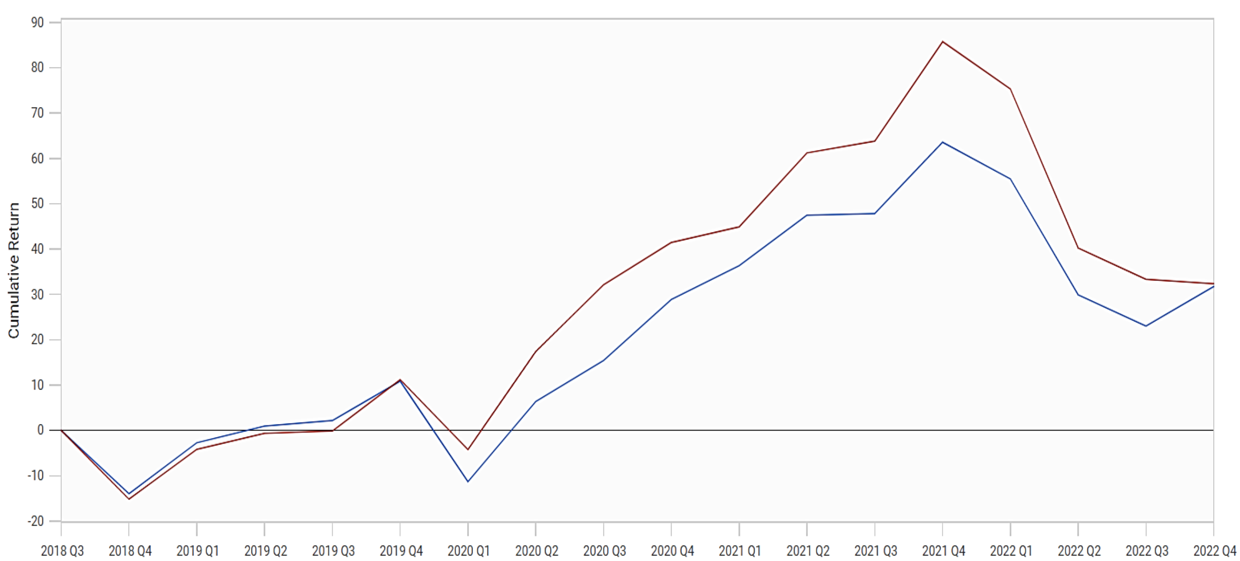

Figure 4.5: BlackRock Performance Chart

BlackRock was able to help many companies during the Global financial crisis of 2008.

The federal reserve hired BlackRock to protect their investments against covid related recessions in recent events. [25]

They used this to grow even further allowing unprecedented power over not just the economy but over the world.

Due to all this power people talk about how the CEO could control all the trends, influence online changes etc. and with helping multiple cooperations via their investment protection and other such services it oversees 10% of assets of the world's global economy.

This would be all thanks to a program named ALADDIN that has access to data due to owning each of the companies such as: banks insurance companies, social media giants, and even pension funds. [28]

Aladdin includes vast amounts of personal data on everyone who has either given/received money via BlackRock.

They also own majority of vanguard, another investment company with people in fox news they control 18% of people there 16% of CBS, 13% of comcast which owns NBC, MSNBC, CNBC, and sky news, they even own 12 % of Disney.[25]

Blackrock has control of 90% of the media, it may seem as if you have not heard of them yet, that would be because they want to not show themselves, if people find out there is a company with such power and control, they would gain unnecessary attention and ask about how they came to power what all consequences for the power and such might arise etc.

They have also recently made connections with China allowing them to process private human data, the application used was HIKI VISION.[27]

In the current market they are purchasing property causing the cost of property to skyrocket and able to make even more money via mortgaging each property.

11. Conclusion

We can conclude that the tests conducted, data reviewed, and information that was researched has been fact checked and previously had not been recorded or collected in this fashion. Progressing through the data we see how there positive and negative impacts on the stock market have been such being: COVID 19, Russia's political issues including war, etc. However, the positive highlight being BlackRock that was able to progress exponentially.

12. Evaluation and Discussion

Initial explanation of the different models of AI was for context and introduction as we are unable to take further steps if the common person is unable to understand or identify what is taking place.

We discuss about ANN, GAN, ARIMA and LTSM, however instead of going into each of the models and further processing the code, I chose LTSM which is the most popular and efficient model evident from most of the research done, hence was selected to be based the program towards.

There was some information on the impact on stock market however there was no clear indication of collection of data as done in this paper, reviewing a working application for code, being able to see the impact that affected throughout the world.

After gathering information from various sources we are able to come to a steady conclusion that there are multiple parameters, both affecting positively and negatively towards the stock market. Positives include investors supporting trusted business, one such example being BlackRock and one of the negative effects being COVID-19.

12.1. Limitations: Due to time constraint and limitations on the monetary side, was unable to progress the research any further.

12.2 Future goals: As presented, main goal was to indicate how the prediction occurred, a good future goal would be to include real-time prediction with highly accurate data.

13. Relevant sources

- Link to GitHub repository: *https://github.com/smarkie?tab=repositories* *(Not published to avoid any copyright/plagiarism issues when grading the work) will make it public once report is graded. (Would not like to take the risk.)*

14. References:

1. Time Series forecasting (no date) Papers with Code. Available at: https://paperswithcode.com/task/time-series-forecasting (Accessed: January 5, 2023).
2. Shen, J. and Shafiq, M.O. (2020) Short-term stock market price trend prediction using a comprehensive deep learning system - journal of big data, Springer Open. Springer International Publishing. Available at: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-020-00333-6 (Accessed: January 5, 2023).
3. Partition data for cross-validation - MATLAB - MathWorks (no date) Available at: https://www.mathworks.com/help/stats/cvpartition.html (Accessed: January 5, 2023).
4. Chopra, R. and Sharma, G.D. (2021) Application of artificial intelligence in stock market forecasting: A Critique, review, and research agenda, MDPI. Multidisciplinary Digital Publishing Institute. Available at: https://www.mdpi.com/1911-8074/14/11/526 (Accessed: January 5, 2023).
5. Adisakshya (no date) Adisakshya/time-series-analysis: Understand and predict the behavior of dynamic systems from experimental or observational data., GitHub. Available at: https://github.com/adisakshya/time-series-analysis (Accessed: January 5, 2023).
6. Machine Learning - Projects Fall 2017 (no date) CS229. Available at: https://cs229.stanford.edu/proj2017 (Accessed: January 5, 2023).
7. ShaneShane 12.1k1717 gold badges7272 silver badges8989 bronze badges et al. (1957) How to make a time series stationary? Cross Validated. Available at: https://stats.stackexchange.com/questions/2077/how-to-make-a-time-seriesstationary (Accessed: January 5, 2023).
8. https://drive.google.com/file/d/1Esfht8c5zDrmKPL9GZJS3MKc882iIPDw/view?usp=sharing
9. M. Berndtsson, J. Hansson, B. Olsson, B. Lundell (2008) - Thesis Projects.
10. Kevin Wayne and Robert Sedgewick (2010) – Algorithms.k
11. Legg, S., Hutter, M. (2007) - Universal Intelligence: A Definition of Machine Intelligence. Minds & Machines.a
12. Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar - Foundations of Machine Learning, 2nd Edition i

13. Richard S. Sutton, Andrew G. Barto - Reinforcement Learning, second edition: An Introduction o
14. Chen, J.- (2022) What is the stock market and how does it work? Investopedia. Available at: https://www.investopedia.com/terms/s/stockmarket.asp (Accessed: November 13, 2022).k
15. Mastodon, R.S. - (2022) Did twitter blue tweet just cost Eli Lilly billions? yes. pic.twitter.com/w4rtjwgcvk, Twitter. Available at: https://twitter.com/rafaelshimunov/status/1591133819918114816?ref_src=twsrc%5Etfw%7Ctwcamp%5Etweetembed%7Ctwterm%5E1591133819918114816%7Ctwgr%5E1a4ebe2f930fa86230947cb2a68d698f1aabd3d0%7Ctwcon%5Es1_&ref_url=https%3A%2F%2Fembedly.forbes.com%2Fwidgets%2Fmedia.html%3Ftype%3Dtext2Fhtmlkey%3Dcfc0fb0733504c77aa4a6ac07caaffc7schema%3Dtwitterurl%3Dhttps3A%2F%2Ftwitter.com%2Frafaelshimunov%2Fstatus%2F1591133819918114816image%3Dhttps3A%2F%2Fi.embed.ly%2F1%2Fimage3Furl3Dhttps253A252F252Fabs.twimg.com252Ferrors252Flogo46x38.png26key3D8804248494c144f5b4765c41f66c6ed5 (Accessed: November 13, 2022). e
16. Figma -The Collaborative Interface Design Tool. (no date) Figma. Available at: https://www.figma.com/ (Accessed: January 15, 2023).N https://www.figma.com/file/cXmJCaUXvmHP7id73G7GaK/Untitled?t=MUzIYWASAwm4LH2s-0
17. Sulandari, W. Suhartono, Subanar and Rodrigues, P.C. 2021. Exponential Smoothing on Modeling and Forecasting Multiple Seasonal Time Series: An Overview
18. Staffini, A. (2022) *Stock price forecasting by a deep convolutional generative Adversarial Network*, *Frontiers*. Frontiers. Available at: https://www.frontiersin.org/articles/10.3389/frai.2022.837596/full (Accessed: April 29, 2023).
19. Cicada (2021) *Share price prediction using Python & Recurrent Neural Nets*, *myBlog*. myBlog. Available at: https://cicada.hashnode.dev/share-price-prediction-using-python-and-recurrent-neural-nets (Accessed: April 29, 2023).
20. *2020 Russia–saudi arabia oil price war* (2023) *Wikipedia*. Wikimedia Foundation. Available at: https://en.wikipedia.org/wiki/2020_Russia%E2%80%93Saudi_Arabia_oil_price_war#cite_note-6 (Accessed: April 29, 2023).

21. *The unprecedented stock market impact of Covid-19 - NBER* (no date). Available at: https://www.nber.org/system/files/working_papers/w26945/w26945.pdf (Accessed: April 29, 2023).
22. *Muhasebe ve FİNANS incelemeleri DERGİSİ - dergipark* (no date). Available at: https://dergipark.org.tr/en/download/article-file/1024340 (Accessed: April 29, 2023).
23. Yahoo Finance - Business Finance Stock Market News, https://in.finance.yahoo.com (Accessed on April 28, 2023)
24. *Blackrock in the U.S.: About Blackrock* (no date) *BlackRock*. Available at: https://www.blackrock.com/us/individual/about-us/about-blackrock#:~:text=BlackRock%20is%20one%20of%20the,and%20supporting%20our%20local%20communities. (Accessed: April 29, 2023).
25. *Blackrock portfolio: Blackrock Inc 13F Holdings & Trades* (no date) *HedgeFollow*. Available at: https://hedgefollow.com/funds/BlackRock (Accessed: April 29, 2023).
26. *Behind the scenes: The corporate governance preferences of ...* (no date). Available at: https://onlinelibrary.wiley.com/doi/10.1111/jofi.12393 (Accessed: April 29, 2023).
27. *Opinion | blackrock's china blunder* (2021) *The Wall Street Journal*. Dow Jones & Company. Available at: https://www.wsj.com/articles/blackrock-larry-fink-china-hkex-sse-authoritarianism-xi-jinping-term-limits-human-rights-ant-didi-global-national-security-11630938728 (Accessed: April 29, 2023).
28. *Blackrock Aladdin – portfolio management software* (no date) *BlackRock*. Available at: https://www.blackrock.com/aladdin (Accessed: April 29, 2023).
29. *Machine learning approaches in stock price prediction: A ... - iopscience* (no date). Available at: https://iopscience.iop.org/article/10.1088/1742-6596/2161/1/012065 (Accessed: April 29, 2023).
30. Shen, J. and Shafiq, M.O. (2020) *Short-term stock market price trend prediction using a comprehensive deep learning system - journal of big data, Springer Open*. Springer International Publishing. Available at: https://journalofbigdata.springeropen.com/articles/10.1186/s40537-020-00333-6 (Accessed: April 29, 2023).
31. Lin, Y. *et al.* (no date) *Improving stock trading decisions based on pattern recognition using machine learning technology, PLOS ONE*. Public Library of Science. Available at:

https://journals.plos.org/plosone/article?id=10.1371%2Fjournal.pone.0255558 (Accessed: April 29, 2023).

32. Cicada (2021) *Share price prediction using Python & Recurrent Neural Nets, myBlog*. myBlog. Available at: https://cicada.hashnode.dev/share-price-prediction-using-python-and-recurrent-neural-nets (Accessed: April 29, 2023).
33. *Machine learning approaches in stock price prediction: A ... - iopscience* (no date). Available at: https://iopscience.iop.org/article/10.1088/1742-6596/2161/1/012065 (Accessed: April 30, 2023).
34. Hochreiter, Sepp, and Ju rgen Schmidhuber. ”Long short-term memory.” Neural computation 9.8 (1997): 1735-1780.
35. Zhou, Peng, et al.” Attention-based bidirectional long short-term mem- ory networks for relation classification.” Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2016.
36. Srivastava, Nitish, et al.” Dropout: a simple way to prevent neural networks from overfitting.” The journal of machine learning research 15.1 (2014): 1929-1958.
37. Pham, Vu, et al.” Dropout improves recurrent neural networks for hand-writing recognition.” 2014 14th International Conference on Frontiers in Handwriting Recognition. IEEE, 2014
38. Connor, J.T.; Martin, R.D.; Atlas, L.E. "Recurrent neural networks and robust time series prediction," IEEE Transactions on Neural Networks, Vol. 5, Iss. 2, pp. 240-254, March 1994.
39. George E. P. Box; Gwilym M. Jenkins; Gregoly C. Reinsel, Time Series Analysis, Prentic; Hall, 1994.
40. James A. FreemanDavid M. Skapura, Neural Networks Algorithms, Application, and Programming Techniques,Addison Wesley 199 1.
41. Gia-Shuh Jang; Feipei Lai; Bor-Wei Jiang; Li-Hua Chien, "The intelligent trend prediction and reversal recognition system using dual-module neural networks," Proceedings. The First International Conference on Artijkial Intelligence on Wall Street, Vol. 26, No. 10, May. 1987.
42. Yu, E. S.; Chen. C.Y.R., "Traffic prediction using neural networks",IEEE Global Telecommunications Conference, including a Communications Theory Mini-Conference. vol. 2. p.991-5, Dec 1993.
43. *Forecasting of Indian stock market using time-series Arima Model | IEEE ...* (no date). Available at: https://ieeexplore.ieee.org/abstract/document/6970973 (Accessed: April 30, 2023).

44. *Stock market trend prediction using Arima-based neural networks | IEEE ...* (no date). Available at: https://ieeexplore.ieee.org/document/549236 (Accessed: April 30, 2023).
45. Author links open overlay panel Yinghao Ren a, a and Abstract With the in-depth study of the stock market (2020) Impact of news on the trend of stock price change: *An analysis based on the deep bidirectional LSTM model, Procedia Computer Science. Elsevier.* Available at: https://www.sciencedirect.com/science/article/pii/S1877050920315829 (Accessed: April 30, 2023).
46. *Global Markets one year after Russia's invasion of Ukraine* (no date) *MSCI*. Available at: https://www.msci.com/www/blog-posts/global-markets-one-year-after/03668219477 (Accessed: April 30, 2023).

15. Appendix

Error: Keras import not working; to patch add .python to fix as shown below.

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import pandas_datareader as web
import yfinance as yf
import datetime as dt

#Patched bug. Noted.
from sklearn.preprocessing import MinMaxScaler
from tensorflow.python.keras.models import Sequential
from tensorflow.python.keras.layers import Dense, Dropout, LSTM
```

Figure X

Failed Test #2 causing highly inaccurate results due to faulty algorithm.

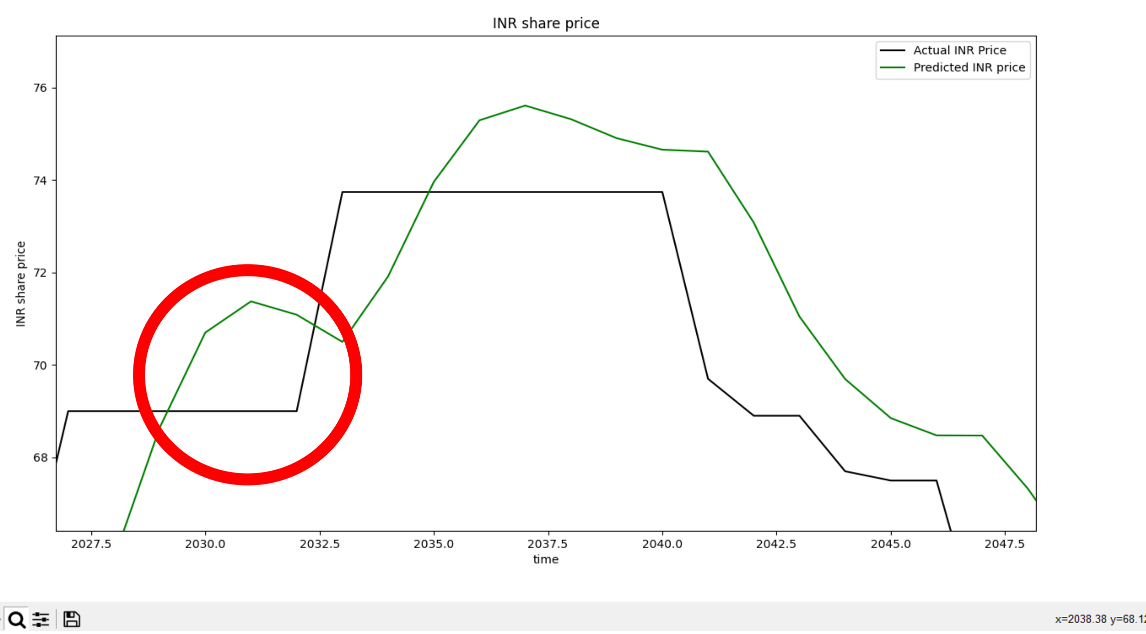

Figure Y

Algorithm not included due to faulty, inaccurate results. True Value: 69.0427

Predicted value: 73.892