Stock Market Price Prediction Using LSTM

Moiz Ullah Khan CS172061

Muhammad Qasim CS172029

Haider Naseer CS172041

I. Abstract

Short-term price movements make up a significant portion of the securities markets' uncertainty. Predicting market movements in the stock market with accuracy is a big economic benefit.

Machine Learning is growing day by day because the problems never end. In the previous years we are using traditional programming concept for solving the problems. Now with the passage of time problems are becoming advance so the technologies to the solve the problem are also become advance. Machine Leaning are growing in health sector, business, climate change and classification-based problems.

Now we are moving towards our topic which is related to business. In this paper we are discussing the hot favorite top of industry which is stock price prediction. Researches and developers are exploring this topic from year 2010 till now. We are discussing different aspect of machine learning and the methodology we are using.

The above process is usually accomplished by doing a basic examination of the organization. Another technique, which has recently been subjected to a number of evaluations, is to use AI to create an intelligent algorithmic behavior predicted model. To teach robots to make purchases and sales decisions in such a short period of time, the final technique must be adopted. size-able. The most spectacular advancement in system learning, Neural Networks, has been used to propel a flitting assumption variation. Specific analysis appropriates the structure to obtain the styles from the actual costs managed into it, as well as attempting to empirically measure the stock's concise future charges. The research appears to be focused on two distinct types of artificial neural networks: feed forward and backpropagation neural networks. The analysis demonstrates that Feed Forwards Multilayer Perceptron's are superior to long short-term memories when it comes to anticipating a stock's short-term fees.

II. Introduction:

As I said earlier in this paper that the stock price is the hot favorite top of industry. Why is it? Because when any person put money in business, they want return. Similarly in stock market when you take share of 10,000 rupees (PKR. per share 20rs) you are expecting the return of 10,000 are more. Machine Learning are now going to easier this by predicting the stock price. Predicting the stock market and take the decision whether you buy more share or sell it.

In this paper we are discussing the methodology. We take the data from Kaggle which is the best platform for Artificial Intelligence, Machine Learning and Deep Learning practitioners. We have 13 years of Pakistan stock market from 2008 till 2021(February).

Forecast of future motion of inventory costs has been the difficulty be tallied of many exploration paintings. On one hand, we have advocates of the green market hypothesis who guarantee that stock costs cannot be predicted. Alternatively, there are paintings which have shown that, if as it should be demonstrated, inventory costs can be assessed with a honestly practical level of exactness. The ultimate has zeroed in on inclination of factors, reasonable pragmatic types and strategies of forecasting. The neural corporations utilized in those capacities have additionally developed and multiplied due to the vertical push of profound mastering. for instance, assist acquiring records on has received ubiquity on the grounds that AlphaGo crushed the quality chess participant at the time via its use, and help turning into greater familiar with has been completed inside the financial expectation situation because of the way that then. these revolutionary jump forwards have given the stock and foreign exchange expectation fashions a stable established order to start and better space to improve. The mainly complex nonlinear courting of profound studying can absolutely paint the muddled qualities of the impacting factors. Several extraordinary fields have exhibited the precision of a profound obtaining records on model for forecast exactness, as an instance, image grouping and fine research. exam results also are obtained for time-association statistics research and expectation with a profound thinking about calculation; for instance, profound turning into more familiar with is utilized to foresee disconnected hold visitors. Normally, profound considering fashions have incredible exhibitions in numerous question fields. sooner or later, it's miles possible to expect inventory and foreign exchange advancements with profound getting to know. financial professionals all for the duration of the planet had been perusing and inspecting the changes inside the stock and foreign exchange markets. The widening application of engineered ability has caused a developing wide collection of purchasers utilizing profound thinking about version to expect and look at stock and forex fees. it's been tried that the change in inventory and foreign exchange value should be expected. no longer similar to ordinary factual and econometric models, profound inspecting can depict complex impacting factors.

III. Related Work:

Neural Networks

(A): Convolutional Neural Network (CNN)

CNN was once comprehensively utilized within the difficulty of image consciousness due to the truth of its compelling instance center ability; its usage became likewise stretched out to the space of monetary forecast. just like the ordinary neural enterprise, CNN is comprised of numerous neurons related via making use of a modern creation, and the loads and predisposition between layers may be prepared. CNN is exquisite from the corporation nation of a totally linked nearby area like profound short organization (DBN), Sparse Autoencoder (SAE), backpropagation (BP), as the CNN can proportion the load many of the neurons in every layer of the organization. as a result, the model extensively decreases the heaviness of the agency and tries now not to fall into dimensional disaster and neighborhood minimization. within the occasion that the attributes of the inventory marketplace at a particular time factor are considered as an element chart, CNN has the viable to eliminate the characteristics of the stock market at the evaluating period from those trademark diagrams. Thusly, CNN can be applied to manufacture a planning willpower existence sized version and can at ultimate be utilized to complete the structure of the circumstance choice system.

(B): Recurrent Neural Network (RNN)

RNN has an area with the neural employer, and it's far desirable at demonstrating and coping with successive information. the exact articulation is that the RNN is professional to maintain the former country, and the past kingdom may be applied within the current realm estimation. The stand-out Appl. Syst. Innov. 2021, 4, 9 three of 30 secret layers are non-self-sufficient, and the contribution of the modern secret layer accommodates of no longer, at this factor simply the yield of the enter layer but similarly the yield of the as of lately blanketed up layer. subsequently, RNN has an appropriately execution in coping with successive records. The advantage of RNN is that An it thinks about the placing of data within the system of making ready, which is really reasonable for the scenario of offers and forex in mild of the fact that the variance at a selected time often conveys some affiliation with the former pattern.

(C): Long Short-Term Memory (LSTM)

Start Paraphrasing we practice a massive learning process of break trust the use of LSTM associations. LSTM might be an assortment of Tedious Neural Frameworks (RNNs) - neural associations with enter circles. In such associations abandon at the modern-day time portion relies upon upon the cutting-edge duties reasonable as previous country of the organization. Notwithstanding, LSTM defeats the issue of vanishing and detonating slopes of RNNs in the midst of backpropagation in studying the hundreds of the affiliation joins. We use Python programming tongue and TensorFlow significant studying machine for executing a LSTM affiliation likewise, use the association to count on the cease achievable good points of protections trade.

Comparing different Models:

Neural networks are a great creation in the era of deep learning. Basically, it is the replication of human neurons. The similar architecture is imposed in neural networks. The way the human brain fires the signal to the neurons for activating it. In human systems different neurons behaving in different manner similarly different neural network models architecture acts in different manner either its is CNN or the modern neural network like BERT.

There is not a lot of change in the architecture of neural network. They just do the simple changes in the architecture and the neural network works differently.

There are different neural networks like CNN, RNN, FRNN, Bert and etc. But our review is on the Convolutional Neural Network, Recurrent Neural Network and Long Short-Term Model (variant of RNN).

Now, we do the comparison between the convolution neural network and recurrent neural network.

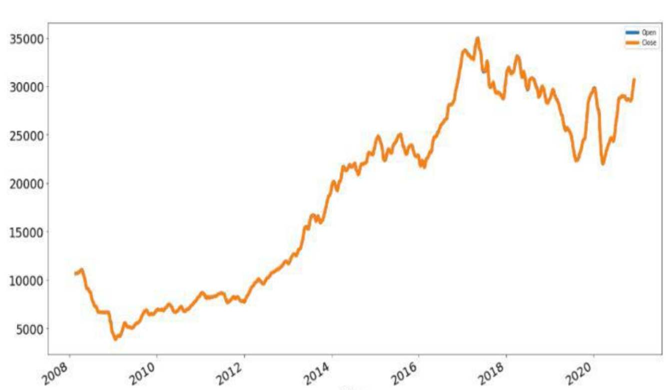
On Convolutional neural network there are two people who worked on it first one is Hubel and the other is Wiesel. They both worked on it in the period of 1950 to 1960. The aim to worked on that is basically the image virtualization and image vision. As everyone see there is a lot of improvement in the image vision and classification area but it does not mean that every model used for different purposes and give us the good accuracy. CNN architecture is basically structured for the image classification and image visions. There are multiple layers in it. The first one is input layer; middle layer are called hidden layer and the last one is output layer. In the middle layer which is called hidden layer there are multiple hidden layers which basically connected to the set of different neurons.

As I have already highlighted that the CNN architecture is for image vision and classification because it consumes a lot of data approximate 3 to 4 weeks of data but it does not guarantee that the model gives good accuracy. While on the other hand Recurrent Neural Network is basically have somehow the similar architecture. But there is a different in the recurrent neural network. Recurrent Neural Network is structed for sequence and series problem. They structed in two ways feed forward and backpropagation approach. They called it predictive modeling. Predictive models are basically get the sequence or series of data and perform their training and it gives approximate good result in the stock prediction because stock price is basically the same as the sequence they get the data into batches and perform it calculation and on the basis of loss they updated their values of weight and biases.

RNN performs well on the stock prices but it cannot perform accurate on the stock prices. On the other hand, this raises the problem in the industry now they moved to make some change in the RNN and they called it Long Short Term Model (LSTM). LSTM have almost the same architecture but they changed it in the activation functions to calculate the loss more accurately. Now LSTM have perform well in the stock market and in the digital platforms as well like Bitcoin mining and all of the stuff like that. This is the mostly used model for stock prediction in the modern world.

Our Related Work

We start by taking data and visualizing it with pandas, matplotlib and seaborn library. We discovered the Box-plot since most people don't care about statistics, don't correlate things, don't grasp data, and most newcomers make these mistakes, which is why we visualize the data. By visualizing the data, we discovered that the stock market crashed in 2009. As a result, we believe that an algorithm should be developed to identify when the market crashed and to find the links related to the cause of the crash and display them on the dashboard, making it easier for users to determine why the market crashed. The algorithm will also assist users in finding market-moving news.



,

```
#searching for the market crash in 2008
from googlesearch import search
query = 'Pakistan stock market crash 2008'

print('Links for market crash:')
for j in search(query, num=5, stop=5):
    print(j)
```

```
Links for market crash:
https://www.dawn.com/news/1196402
https://www.dawn.com/news/686687/revisiting-stock-market-crash
https://www.thenews.com.pk/print/55921-market-crash-2008-pinned-on-secp-ex-chairman-kse-broke
https://tribune.com.pk/story/940353/investigating-the-2008-crash
```

We divide the data into training and testing groups before normalizing it. After that, we give the model the training data. We establish several layers in the model, ranging from little neurons to larger neurons, and these layers will gradually reduce, eventually giving the output on a single neuron. Because our model predicts the stock market's closing price, we'll only have one neuron in our output layer. With the use of data in the form of vectors that travel through the model, the model will calculate the loss. Because LSTM (Long Short-Term Model) is a kind of recurrent neural network, it will repeatedly execute this operation until it concludes that this is the minimal loss and these are the optimal weights. We performed 100 epochs to train our model. Finally, we run tests to estimate the stock market's closing price.

IV. Dataset

13 years of statistics with date, opening, closing index, High and close price, change and total volume were mentioned in the dataset. We have the data from 2008 till 2021(February). As we have already stated above that our aim is to develop the framework to predict the stock closing prices of Pakistan stock Exchange (PSx-100). Every element which has been using in the dataset define its own characteristics. The elements are defined in the following:

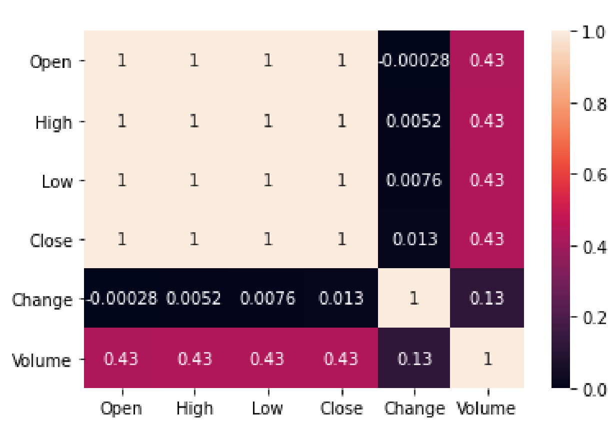
(I) Date
(ii) Open Price

(iii) Closing Price
(iv) High
(v) Low
(vi) Volume of market

Date	Open	High	Low	Close	Change	Volume
2021-02-23	31722.16	31800.90	31597.31	31626.19	-21.38	718191025
2021-02-22	31874.78	31958.58	31612.55	31647.57	-203.61	721952658
2021-02-19	31748.75	31904.30	31749.43	31851.18	91.36	694795084
2021-02-18	32049.85	32104.67	31745.72	31759.82	-288.86	577837595
2021-02-17	32166.21	32390.77	32044.01	32048.68	-93.15	701658181

Figure-2

Making use of the six factors within the crude statistics, we usually have a tendency to deduce the following elements that we have a tendency to utilize quick for constructing our prescient fashions. we usually have a tendency to utilized methodologies for decisive the top price of market. After reading the data we apply some exploratory data analysis EDA in order to get the hidden pattern of data. we basically uses the matplotlib and seaborn for data visualization. We take the correlation between the variables to check the dependencies over one another.



By using the boxplot, we found that the volume column of data contains the outliers so we take the quantile q1 and q3 to find out the inter quantile range. After inter quantile range we discard the data less than 25% and above 75% by using the or-bitwise operator. We again plot the boxplot by using seaborn. We somehow decreased the number of outliers in our dataset.

After all this now we have much understanding about data. We split the data into two measure parts. The first one is the training set and the other is test set. We scale the data by using the minmax scalar. clean textual content Output Be that due to the fact it'd, we have a propensity to teach the corporation with particular age esteems and cluster sizes of the facts beneath numerous times and for several shares to build up the first-rate exhibition of the enterprise.

Boxplot Before removing outliers

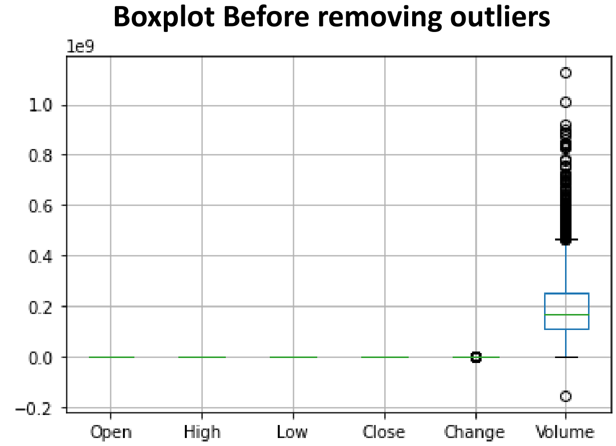


Figure-3

Boxplot after removing outliers

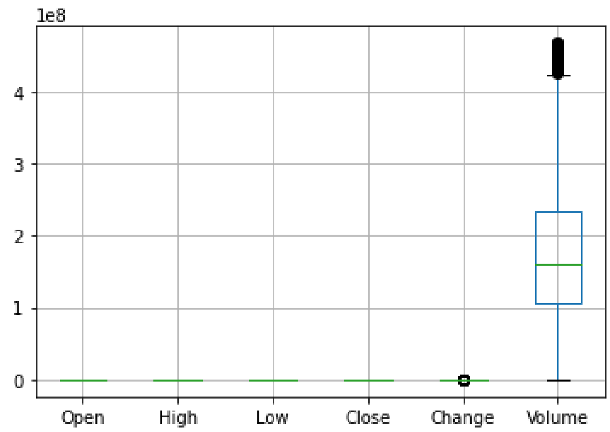
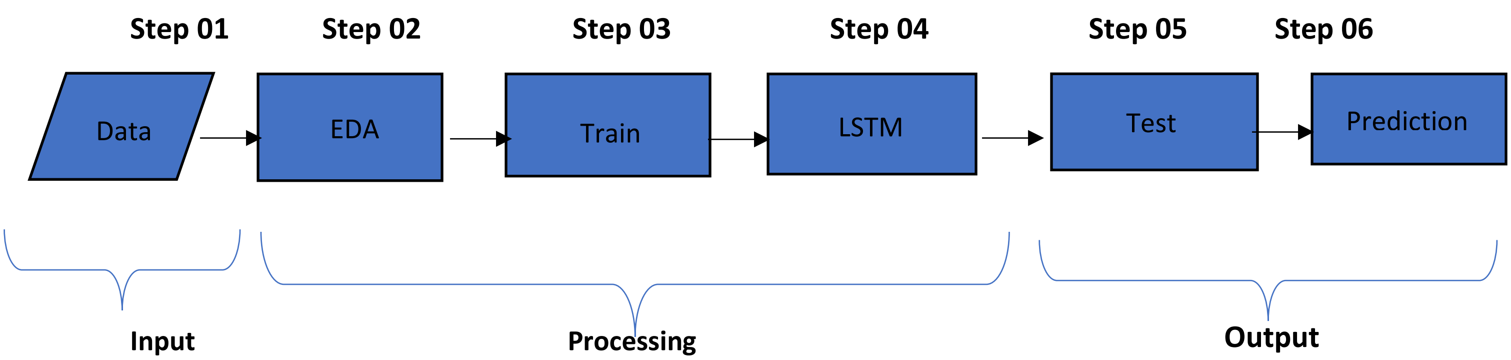


Figure-4

V. Methodology:

Here is the complete workflow of how we are doing the things. There are six steps we have done in this project and they are divided into further parts also. Discussing the first Step which is Data.

Workflow of Our Methodology



Data:

13 years of statistics with date, opening, closing index, High and close price, change and total volume were mentioned in the dataset. We have the data from 2008 till 2021(February). As we have already stated above that our aim is to develop the framework to predict the stock closing prices of Pakistan stock Exchange (PSx-100). Every element which has been using in the dataset define its own characteristics. The elements are defined in the following:

(I) Date
(ii) Open Price
(iii) Closing Price
(iv) High
(v) Low
(vi) Volume of market

Date	Open	High	Low	Close	Change	Volume
2021-02-23	31722.16	31800.90	31597.31	31626.19	-21.38	718191025
2021-02-22	31874.78	31958.58	31612.55	31647.57	-203.61	721952658
2021-02-19	31748.75	31904.30	31749.43	31851.18	91.36	694795084
2021-02-18	32049.85	32104.67	31745.72	31759.82	-288.86	577837595
2021-02-17	32166.21	32390.77	32044.01	32048.68	-93.15	701658181

Figure 01

By using these six elements within the dataset. After the whole process of cleaning, we use these seven factors to predict the stock closing price. We use the regression approach in order to predict the closing price. We are moving towards our next step which is Exploratory Data Analysis – EDA.

Step 02:

Exploratory Data Analysis – EDA:

EDA which is the abbreviation of is Exploratory Data Analysis – EDA. After reading the data we apply some exploratory data analysis EDA in order to get the hidden pattern of data. We basically use the matplotlib and seaborn for data visualization. We take the correlation between the variables to check the dependencies over one another.

As the expert says that Exploratory Data Analysis is the most crucial part before implementing the model. Because the model is work on data and if the data is not accurate then our model is not accurate to predict the price. So there are steps in order to analysis, cleaning and process the data for getting the accurate results. The first steps we have to do is getting the information about the columns. So we have we use info function:

```
st_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 3221 entries, 2021-02-23 to 2008-0
Data columns (total 6 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   Open    3221 non-null   float64
 1   High    3221 non-null   float64
 2   Low     3221 non-null   float64
 3   Close   3221 non-null   float64
 4   Change  3221 non-null   float64
 5   Volume  3221 non-null   int64
dtypes: float64(5), int64(1)
memory usage: 176.1 KB
```

Figure 02

This function tells us about the complete information about data. The first the things we have found the total number of columns, associated the non-Null column. This defines the total non-null values in our columns. This function also defines the data type in front of each column.
Now with this function we have found that we have to make the date column as index. So, we know the everyday price of stock.

Data Preparation:

As you can see the above image the date column type is object so we need to change the type by using pandas library function to_datetime () and then set date as index. Every column has object type so change the type of columns before moving onwards.
Datatype code figure

We have use to numeric so they can automatically get the suitable type for every column. The open, high, low, close and change column are float type because they contain floating point values and volume column data type is integer.

Removing comma in prices:

After changing the data type, we use head function to visualize the data. There are commas in the prices. So, we use replace function with regex parameter. Now our data have enough capable to do things.

We are finding that our data has null values or not. We use isnull function to get the total null values in each column of dataset. We remove the duplicates in our dataset because we have to away from overfitting the model. By dropping duplicates rows we use drop_duplicate function.

Visualization, Correlation and Boxplot:

Now our data have enough to get the hidden patterns and visualize some results. we basically uses the matplotlib and seaborn for data visualization.

```
#correlation b/w columns/ variables
correlation = st_df.corr()
```

```
#finding the correlation b/w variables
sns.heatmap(correlation, xticklabels=correlation.
            yticklabels=correlation.columns,
            annot=True)
```

```
<AxesSubplot:>
```

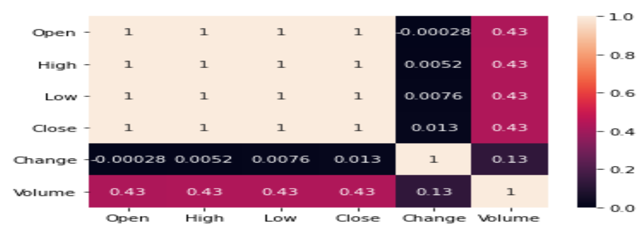


Figure 03

We take the correlation between the variables to check the dependencies over one another. In the above diagram the dark color defines the negatively correlated and the light color shows the high correlation among variables. Open column is correlated with close variable. Open, high, close, low have same correlation which is 0.43 with volume. Soo, its mean that volume is depending on these factors. Change variable also correlated with volume variable by 0.13.

Boxplot:

```
st_df = st_df[~((st_df < (Q1 -1.5 * IQR)) | (st_df > (Q3 + 1.5 *IQR
```

```
st_df.boxplot()
```

```
<AxesSubplot:>
```

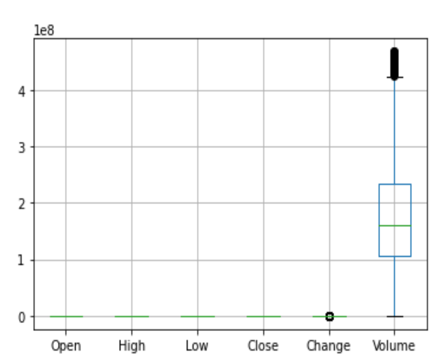


Figure 04

We have used the describe function to get the description of data. When we compare the 75 percentiles with the max column there is a lot of difference between them. So, its mean there are outlier exits in our dataset. By using the boxplot, we found that the volume column of data contain the outliers so we take the quantile q1 and q3 to find out the inter quantile range. After inter quantile range we discard the data less than 25% and above 75% by using the or-bitwise operator. We again plot the boxplot by using seaborn, we somehow decreased the number of outliers in our dataset.

Visualization plays the important role for getting the better representation of data. We plot multiple visualization but we will highlight some visualization which are important for us.

```
#visualize the volume of stock price
st_df['Volume'].plot(figsize=(15,5))
```

```
<AxesSubplot:xlabel='Date'>
```

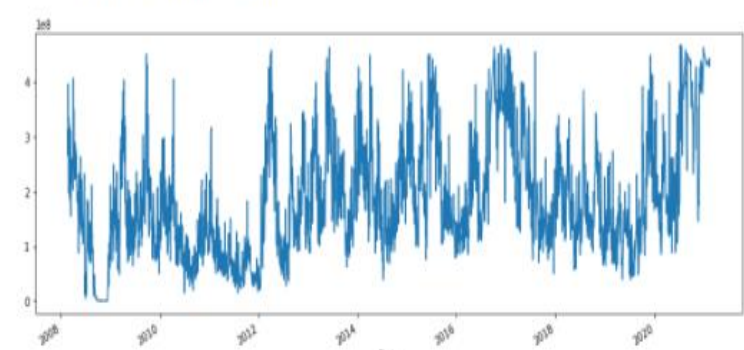


Figure 05

The first we visualize the volume column. We see that the volume column gives us the total volume of our market at a particular day. There is a lot of fluctuation in the volume.

Open close:

The last thing which we have want to share you is the open and close variable representation.

We have seen that the open and close values are almost same but there is a minor difference between these columns.

As you have seen that the stock market is crash in the 2009, we have shared the news so anyone knows that what is the reason behind that.

Figure 06

Now we are moving to make the train and test set. We import some libraries like Keras, sklearn and TensorFlow for multiple purposes. After all this now we have much understanding about data. We split the data into two measure parts. The first one is the training set and the other is test set. We scale the data by using the minmax scalar.

Step 03:
Making Train Set:

For training data set, we take the data and scale the data value by minmax scalar because data have some high and very low values.

Preprocessing the data is essential part so we scale the data in order to normalize it because to highlight the each and every feature of dataset. After that we give the training data
to model for training and the model will train on this dataset. After completing the training of our model, we move towards testing.

```
#scaling the train and test data
# st_df = st_df.reset_index()
scale = MinMaxScaler(feature_range = (0,1))
train_set_scaling = scale.fit_transform(train_set)

train_X = []
train_y = []

for i in range(60, len(train_set_scaling)):
    train_X.append(train_set_scaling[i-60:i, 0])
    train_y.append(train_set_scaling[i, 0])

train_X, train_y = np.array(train_X), np.array(train_y)

train_X = np.reshape(train_X, (train_X.shape[0], train_X.shape[1],1))
```

Figure 07

Step 04:
LSTM Model:

There are several models for predicting the stock prices but the thing is that few peoples know what model to apply on it.

In the era of deep learning, people use convolutional neural networks for predicting the stock market price but the **CNN** almost takes 3 to 4 weeks of data to perform good training while the Long Short Term Model LSTM which we have uses takes the one week of data to give you the best output with good accuracy.

We first initiate the model and add up the first layer of 30 neurons and with 0.3 dropout rate. Similarly, we add the multiple layers of 50,100,50 and 30 neurons.

The output layers contain only one neuron because the predicted price is only one which is closing price. We use the two-activation function just before the output layer and on the output layer. On the second last layer we use rectified linear unit **relu** and on the output layer we use **SoftMax** because **SoftMax** have better understand of value and then backpropagate it by checking the loss.

```
model = Sequential()

#adding the input layer
model.add(LSTM(30, return_sequences=True, i
model.add(Dropout(0.3))

#Dense layers
model.add(LSTM(50, return_sequences=True))
model.add(Dropout(0.3))

model.add(LSTM(100, return_sequences=True))
model.add(Dropout(0.3))

model.add(LSTM(50))
model.add(Dropout(0.3))

model.add(Dense(30,activation='relu'))

#ouput layer
model.add(Dense(1, activation='softmax'))

#calculating loss
model.compile(loss='mean_squared_error', op

# fit network
model.fit(train_X, train_y, epochs=100, bat
```

```
Epoch 93/100
30/30 [==============================] - 3s 95ms/step - loss: 0.24
Epoch 94/100
30/30 [==============================] - 3s 95ms/step - loss: 0.23
Epoch 95/100
30/30 [==============================] - 3s 95ms/step - loss: 0.24
Epoch 96/100
30/30 [==============================] - 3s 94ms/step - loss: 0.23
Epoch 97/100
30/30 [==============================] - 3s 95ms/step - loss: 0.25
Epoch 98/100
30/30 [==============================] - 3s 95ms/step - loss: 0.24
Epoch 99/100
30/30 [==============================] - 3s 97ms/step - loss: 0.24
Epoch 100/100

30/30 [==============================] - 3s 95ms/step - loss: 0.24

<keras.callbacks.History at 0x20d8fc50970>
```

Figure 08

Step 05:
Making Test Set:

For testing data set, we take the data and scale the data value by minmax scalar because data have some high and very low values.

Preprocessing the data is essential part so we scale the data in order to normalize it because to highlight the each and every feature of dataset. After that we give the test data
to model for evaluation and the model will give us good performance. By testing data model predict the closing price of stock market.

```
train_data = st_df.iloc[:800 ,1:2]
test_data = st_df.iloc[800: ,1:2]

df = pd.concat((train_data, test_data), axis=0)

inputs = df[len(df)-len(test_data)-60:].values

inputs = inputs.reshape(-1, 1)
inputs = scale.transform(inputs)
test_X = []

for i in range(60, 519):
    test_X.append(inputs[i-50:i, 0])

test_X = np.array(test_X)
test_X = np.reshape(test_X, (test_X.shape[0], test_
print(test_X.shape)
```

```
(459, 50, 1)
```

Figure 09

Step 06 Final Step:

Predictions:

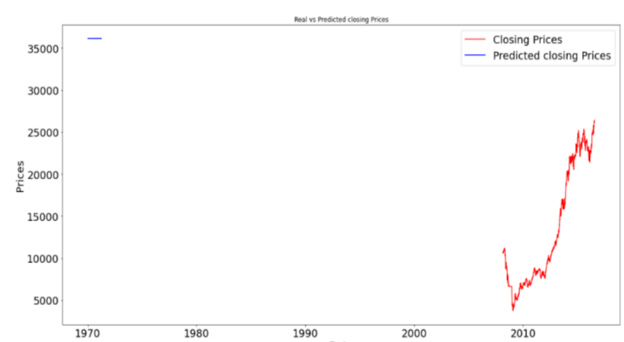


Figure 10

VI. Experiment:

For the PSX-100 index, the experiments compare the suggested method's accuracy to that of a deep learning system. the price estimate and the actual price have a linear relationship. To simplify the time duration, the data is separated into different groups for training and testing. The LSTM is used to forecast the pace of price change. This is the paper's first contribution. One other impact is that we use earlier knowledge about the relationship between prices and the rate of change to improve the model's performance, and the results of the experiment show that using the rate of expansion to implicitly predict the price of stocks is more accurate than directly predicting the price of stocks. The obtainability will be used to explain the relationship between the previous day's price and the next day's price, resulting in increased performance and easier-to-understand results.

VII. Results:

Here is the prediction of our model. The x-axis shows the year and y-axis shows the closing price which our model is predicted. After all that analyzes, we are going to predict the markets closing price. Reinforcement learning does not have the same labelling information as supervised learning in reality, and the results are commonly obtained after attempting the action. As a result, reinforcement learning modifies the preceding strategy in real time based on the results.

We were using LSTM which are a part of Deep learning (branch of Machine Learning). Because LSTM takes one week data for prediction. LSTM gives us good result. The last closing price of stock market hit the 30,000 and our model predicted the closing price of stock market is 35,000 with the loss of 24% percent.

VIII. Conclusion:

There are so many peoples publishing their notebooks and researches but unfortunately

very few of them succeed. In our case, we are not very hard to try predicting the stock price because predict the stock price is very difficult because there are a lot of features like news running in the market, fears of losing, threads and many others things.

Basically, there are 18 major families which are running the Pakistan stock. These families hold 36 to 42 % stocks of Pakistan. Big players play the game of their own rules and run the market. There are also fake news as well for disturbing the stock market.

However, we use the LSTM model for stock price prediction because LSTM takes one week data for prediction. LSTM gives us good result.

There are many algorithms for predictive modeling but we have seen that CNN is used for image classification and image vision, RNN is used for prediction but the things is that the CNN and RNN both requires a large amount of data for prediction. RNN is perform well in prediction but not much accurate. Then LSTM comes, mostly peoples use the CNN before comes to RNN and similar when they know RNN performs well they shift towards RNN. LSTM is a variant of RNN, it takes only one week data for prediction. It comparison to other it quite well and people shifted towards it.

However, we use LSTM and we found that the loss is quite less which is 24 percent. After the whole review, we found that the LSTM is predicting better stock and trading prices rather than any other else model.

References:

1. A New Model for Stock Price Movements Prediction Using Deep Neural Network

https://dl.acm.org/doi/abs/10.1145/3155133.3155202

2. A multi-layer and multi-ensemble stock trader using deep learning and deep reinforcement learning

https://link.springer.com/article/10.1007/s10489-020-01839-5

3. Stock Prices Prediction using Deep Learning Models

https://arxiv.org/abs/1909.12227

4. A Survey of Forex and Stock Price Prediction Using Deep Learning

https://www.mdpi.com/2571-5577/4/1/9

5. NSE Stock Market Prediction Using Deep-Learning Models

https://www.sciencedirect.com/science/article/pii/S1877050918307828

6. Stock prediction using deep learning

https://link.springer.com/article/10.1007/s11042-016-4159-7

7. Deep learning networks for stock market analysis and prediction

https://www.sciencedirect.com/science/article/abs/pii/S0957417417302750

8. Short term stock price prediction using deep learning

https://ieeexplore.ieee.org/abstract/document/8256643

9. A Robust Predictive Model for Stock Price Prediction Using Deep Learning and Natural Language Processing

https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3502624