DEEP LEARNING FRAMEWORK FOR FINANCIAL TIME SERIES PREDICTION USING TECHNICAL INDICATORS AND PRICE DATA

Yasiru Bimanke Serasinghe Pathirana

(209362J)

Thesis/Dissertation submitted in fulfillment of the requirements for the degree Master of Science in Computer Science

Department of Computer Science

University of Moratuwa

Sri Lanka

July 2022

DECLARATION

I declare that this is my own work and this thesis/dissertation[2] does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my thesis/dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: *UOM Verified Signature* Date: 05/07/2022

The above candidate has carried out research for the Master's Dissertation under my supervision.

Name of the Supervisor: Dr. Amal Shehan Perera

Signature of the Supervisor: Date:

ABSTRACT

Financial Time Series prediction is a challenging task due to its dependency in many socio-economic factors. It depends on both quantitative and qualitative factors in a Financial Market. Quantitative factors can be mathematically modeled but qualitative factors are harder to model. Market behavior depends on both micro-economic as well as macro-economic behavior which includes quantitative and qualitative factors on both of them. Therefore modeling and predicting a financial time series has become a challenging task in Big Data Analytics world.

Deep Neural Networks can be identified as a main tool in Big Data Analytics which could solve the above challenge. Long Short Term Memory Units and Gated Recurrent Units in deep neural networks can accommodate memory cells which can store an accumulated memory. This helped to accurately capture the dependencies of the current data point by previous data points. Financial Time Series heavily depends on their predecessors and these concepts managed to capture such relationships.

This research use a combination of LSTM and GRU Units to accurately predict the Index Close Price of Tadawul All Share Index (TASI) and Stock Close Price of five highly tradable stocks in Tadawul Stock Exchange. Open, High, Low and Close Prices as well as Standard Technical Indicators of Stocks and Indices are primarily used to create the model. Principal Component Analysis is used to reduce the dimensionality. OHLC and Technical Indicator Values are fed to the network based on four different topologies creating four Evaluator Models.

DEDICATION

This work is dedicated to my parents who have always loved me unconditionally and whose good examples have taught me to work hard for the things that I aspire to achieve.

This thesis work is also dedicated to my beloved wife who has been a constant source of support and encouragement throughout my life. I am truly blessed for having you in my life.

At the same time I would like to dedicate this thesis for my dear colleagues who have been a constant source helping hand during this post graduate journey. You guys are the best.

ACKNOWLEDGEMENTS

I have received a great deal of support and assistance from many intellectuals during the completion of my dissertation.

First and Foremost, Thank you to my supervisor, Dr. Amal Shehan Perera, for your patience, guidance, and support. I have benefited greatly from your wealth of knowledge and expertise in the field of data science. I am extremely grateful that you took me on as a student and continued to have faith in me over the past year.

I would also like to express my sincere gratitude to Dr. Sapumal Ahangama for the valuable advices provided during the entire research. Your encouraging words and thoughtful, detailed feedback have been very important to me.

Thank You , Mr.Lasintha Ferdinando and Mr. Dassana Jayalath, for your valuable guidance and advices throughout the research period. You provided me with the valuable domain information and insight that I needed to choose the right direction and successfully complete my dissertation.

I would like to extend my thanks towards DirectFN for providing me with all-important data to work on my research. These results and insight would not be possible without the valuable data you provided free of charge.

LIST OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVATIONS

LSTM	Long Short Term Memory
GRU	Gated Recurrent Unit
MAPE	Mean Absolute Percentage Error
TASI	Tadawul All Share Index
OHLC	Open High Low Close

1. INTRODUCTION

1.1 Stock Market

Stock exchange is a market place which connects stock buyers and stock sellers. It does not own any shares but lays a platform for stocks, bonds and other financial instruments to be traded. The Stock Market is known for its volatile and unstable nature [1]. A particular stock could be trending up on one day and trending down on another. Big profits are generated by selling the stocks when they are at their peak price and buying them when they are at their lowest price. One of the main interest in the investors as well as researches is to identify the factors which cause this price movements. Highly accurate stock price predictors can be designed if these causal factors to price changes and their level of impact could be identified. Knowledge on causal factors can lead to maximize profits and minimize losses in a stock market which is worth billions of dollars.

1.2 Market Index

In finance, a stock index, or stock market index, is an index that measures a stock market, or a subset of the stock market. It helps investors compare current price levels with past prices to calculate market performance. It is computed from the prices of selected stocks (typically a weighted arithmetic mean is calculated). Selected stocks belongs to the category which the index will be representative of. There might be a special weighting scheme where each stock will not contribute equally for the index value calculation. Predicting the Market Index will help to understand the market movement on the next date. It will help the investors as well as the exchange to take necessary actions to mitigate risks as well as ride on opportunities.

A market index measures the value of tradable instruments with a specific market characteristics. Index provider decides the procedures and methods of adding index constituents as well as weightage of each constituent. Weightage of a certain tradable instrument within an Index is defined usually by its price or market capital. Investors use the Indices as indicators of overall market behavior. A negative impact on index means that certain market is not performing well and vice versa. [2].

United States have S&P 500, Nasdaq Composite and Dow Jones as their three main indices. These three indexes include the 500 largest stocks, all of the stocks on the Nasdaq exchange and 30 largest stocks in the U.S. by market capital. So the performance of this indices will determine the state of the USA Economy and Market.

Tadawul is the main stock exchange in Saudi Arabia. Tadawul All Share Index (TASI) is the main index in the Tadawul exchange. A model which can predict the TASI Index can predict the movement of Saudi Stock Exchange and the economy as well. It will also give an indication on where the Middle East as a region will move economically since Saudi Arabia is the economic and political giant in Middle East.

2. RESEARCH PROBLEM

Stock Market Prediction is one of the most sought after yet challenging task in the timer series prediction domain. It is simply because of the volatility and noise of the data series. The other reason can be cited as the influence of non-monetary socio-economic macro factors towards stock prices. Investors always look at the stock with a view of the corresponding company. The company policies, long term vision, investment decisions, key personnel changes, social welfare activities etc. will be key in determining the impression and attitude towards the company. This attitude directly correlate with the purchase decision of investors. Because they won't invest their money on companies with bad reputation or no long term vision even if it's trading at the peak price. Other way around if the company is engaging in social welfare activities, treat employees fairly, make joint ventures or acquisitions etc. then the investors will obviously invest their money on that stock even its not profitable at the moment. Identifying those patterns to perfection via numerical analysis is not feasible.

Many researches have tried to unlock the secrets of the price movements in different markets. Artificial Neural Networks (ANNs) [3] as well as the Support Vector Regression (SVR) [4], are mainly used to predict the financial time series and obtain competitive advantage over the others [5][6]. Applying nonlinear topologies have become the recent norm as most the experts believe hidden patterns in time series are mostly nonlinear in nature. The concept has shown advancement in the field of financial time series prediction by being able to capture more complex relationships hidden in the time series [7] and achieve state-of-the-art gains [8]. However dealing with financial time series is still considered as one of the challenging task in the field of machine learning and pattern recognition [9].Research have been conducted with state of the art techniques on different markets specially United States of America and England. These researches have always explored new novel ideas in them and contributed to the advancement of the field in general. But no research has been done related to the Tadawul Stock Exchange in Saudi Arabia.

3. RESEARCH OBJECTIVES

Main objective of the research will be to build a deep learning model which predicts the close price of the TASI (Tadawul All Share Index).

Main Objectives of the research

1. Use of Noise elimination techniques to eliminate noise in financial data series.
2. Identify technical indicators which will impact the close price of the TASI Index.
3. Use raw Price Data with Technical Indicators and Fundamental Analysis to predict the Index Price.
4. Build a Prediction model using Long Short-Term Memory (LSTM) cells and Gateway Recurrent Units (GRU) to predict the close prices of the TASI Index.

Secondary Objectives of the research

1. Identify the level of prediction accuracy of TASI index with the combination of different technical indicators.
2. Identify the level of prediction accuracy of TASI Index with the different combination of feeding the raw price data and technical indicators.
3. Apply the LSTM and GRU Hybrid Prediction Model in Selected Stocks in Tadawul Stock Exchange and evaluate the accuracies.

4. RELATED WORK

4.1 Deep Learning Frameworks for Predicting Financial Time Series

There are four main types of deep learning mechanisms which is more frequently used in predicting the financial time series.

1. Deep Belief Networks
2. Convolutional Neural Networks
3. Stacked Auto Encoders
4. Long Short-Term Memory
5. Gated Recurrent Units (GRU)

4.2 Deep Belief Networks

Deep Belief Network is an alternate type of deep neural network where hidden units in the neural network does not connect to other hidden units of the same layer. But they connects to their preceding and succeeding layer hidden units [10].

Unsupervised training is used to train the DBNs at first. In this phase layers usually capture the patterns and features of the time series. Then DBNs are trained with supervision to make them useful in classification task.

DBNs are usually made up of Restricted Boltzmann machines (RBMs) or auto encoders. Each preceding layer acts as the visible layer to the next layer [10]. RBM is an undirected model with a visible input layer and many hidden layers. Neurons in each layer does not have any interconnections between them. But they all connect to preceding and succeeding layer neurons. This model construction helps unsupervised layer-by-layer training and hence each layer has a powerful ability to capture distinctive features in the time series.

4.2.1 DBN Architectures used in Stock Price Prediction

Xiumin at el. [16] proposed a DBN architecture with use of several Restricted Boltzmann Machine layers and a Back Propagation Layer. He used this DBN with the help of intrinsic plasticity concept of neurons to build a stock price prediction model for S&P500 index.

Figure 4.1 shows the architecture used by Xiumin at el. which shows a visible layer and a hidden layer in each RBM. As the Figure 4.1 shows no Units in the same layer does not interconnect with each other. But fully connects with the units of the other layers.

The have trained each RBN layer in the Figure 4.1 independently and take the binary state of each layer. Propagation of any errors are avoided by isolate training the layers. Then we use contrastive divergence (CD) algorithm is used as a learning rule to training the networks.

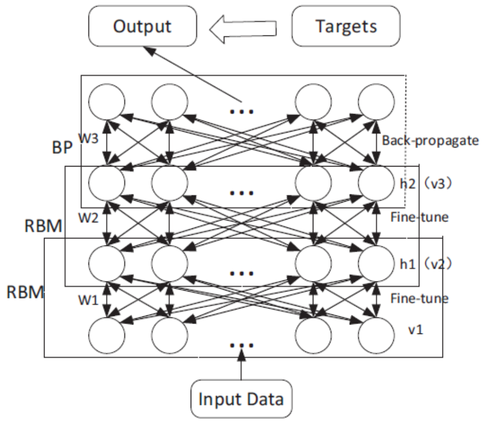


Figure 4.1: Xiumin at el DBN Architecture for Financial Time Series

4.2.2 Intrinsic Plasticity (IP)

Idea is based on the information maximization principle. The synaptic response of a neuron will be maximized when there is a positive response and resistance will be maximized when there is an inhibitory response. This is achieved in models via KL-Divergence principle which has an exponential out function.

$$f(x, a, b) = \frac{1}{1 + \exp(-ax + b)} \qquad (1)$$

X = synaptic input

f (x, a, b) = output response of the neuron.

Parameter ‘a’ of Equation (1) controls the slope of the curve which in return is responsible for the level of discrimination. Parameter ‘b’ of Equation (1) controls the sensitivity of the neuron.

4.2.3 Experimental Set up of Xiaumin at el.

S&P 500 Index data was used for the experiment. As Table 4.1 shows research used 100 records as testing data which amount 10% of the training data set. Research used basic Open, High, Low, Close Prices and Volume of the S&P 500 Index. They used 3 RBM Layers in their model which incorporated Intrinsic Plasticity feature. Model is shown in the below Figure 4.2.

Table 4.1 Data Set Description of Xiaumin at el

	No of Records
Training Set	1163
Testing Set	100

DBN Architecture

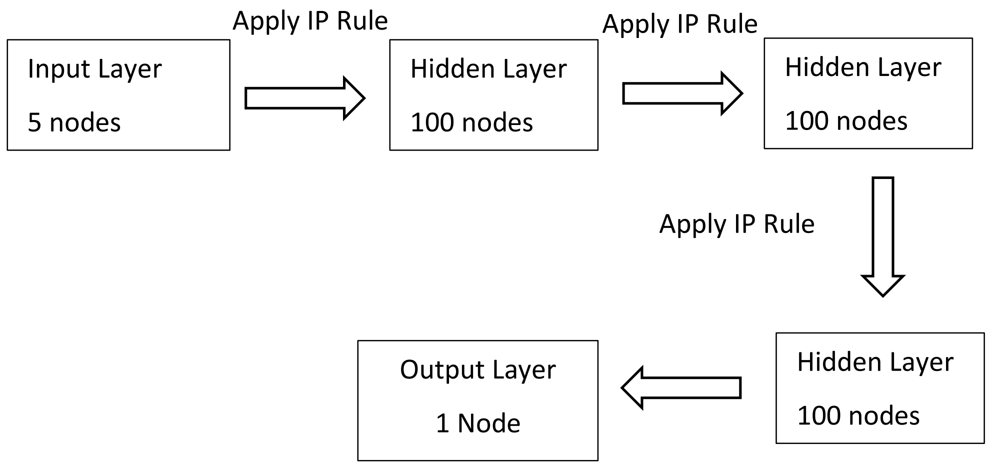


Figure 4.2: Xiaumin at el DBN Architecture

4.2.4 Results of Xiaumin at el.

Below evaluation criteria were used to measure the performance.

1. Mean Squared Error
2. Normalized Root Mean Squared Error
3. Mean Absolute Error
4. Standard Deviation

Figure 4.3 below shows the results obtained by the Xiaumin at el with the research. It shows that both models which applied two types of IP rules deliver better accuracies in terms of Mean Squared Error. Li's Rule applied DBN delivered the best results as per the research. The overall outcome was the application of IP Rule has contributed to higher accuracies than traditional DBN model.

Models	MSE	MAE	NRMSE	SD
LDBN	0.0025145	0.040332633	0.361483367	0.04908427
TDBN	0.0029960	0.043671000	0.406082000	0.06506600
DBN	0.0040088	0.049399400	0.477528530	0.05308217

Figure 4.3: Experimental Results of Xiumin at el

LDBN- Li's Rule applied DBN

TDBN – Trisch's Rule applied DBN

DBN – Traditional DBN

4.3 Convolutional Neural Networks

Dingli at el [11] has researched on use of convolutional neural networks on financial time series prediction task. This model used Tensor flow is a backend which supports training and prediction in CNNs. Softmax regression is used as the primary method of classification which classifies output to a given number of classes.

Dataset preparation to be an input to a CNN is considered an important task in the process. It occupies a significance as CNN is primarily designed and worked with image classification tasks. It uses temporal features to make predictions. Researches have transformed the dataset into a matrix of 8x8, which could incorporate in 64 features.

One-hot vectors were used to retrieve the output from the CNN network. Example can be listed as below.

- 'Up' Movement [1,0]
- 'Down' Movement [0,1]

4.3.1 Architecture of Convolutional Neural Network

First Convolution Layer is computed with 32 features for each 2×2 patch which includes a stride of 1 so that output consists of 2048 (8×8×32) features. In order to avoid any feature dropping during convolution, researchers have used 'SAME' type padding. Each output channel has a bias variable as well. Researchers have applied max pooling to the first convolution in order to reduce the shape to 4×4×32.

Second Convolutional Layer will have 32 inputs instead of 1. This is because second layer need to accommodate the output of the first Layer. Then 64 features are calculated for each 2*2 patch. No feature will be dropped in this layer by adding Padding. Shape of 4×4×64 features will be deduced after applying the convolution and max pooling is used to reduce it down to 2×2×64.

Now reduced image of 2*2 patches will be fed to a fully connected layer and every neuron in this layer will connect to the output of the max pooling layer and output layer.

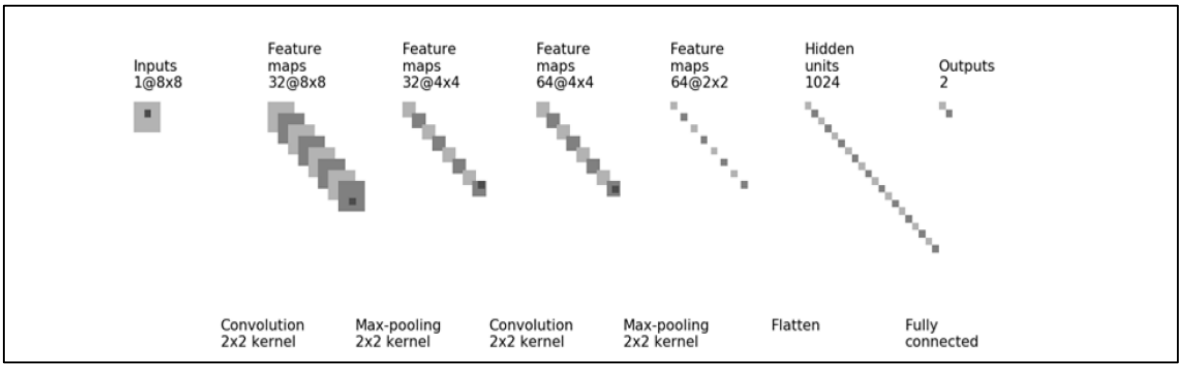


Figure 4.4: CNN Architecture for Financial Time Series Analysis

There are number of steps taken to increase the efficiency and accuracy of the CNN network. These include:

- Network Depth is varied from time to time.
- Learning Rate Variation.
- Local Receptive Area variation.
- Variation of No of features.

4.3.2 Results of the Convolutional Neural Networks

It was noted that accuracy fluctuates from 54% to 64% after performing multiple testing rounds for the monthly dataset and the cause was identified as the random initialization of weights in the CNN layers. So they have averaged out the accuracy of 10 runs as their representative accuracy which is 56%. I found another research focused on optimizing this initial value assignment in hidden layers where they have fully focused on mathematical models to assign optimal values for their hidden layer weights which effects the overall model performance. This 56% accuracy has not been able to surpass leading accurate predictions via MLP which stands around 69%.

Researchers have assumed that the reason for this underperformance was due to the inheriting high complexity of the CNN models. They tried to play with the model parameters and increase the accuracies but did not achieved substantial gains. They obtained an increased accuracy of 62% from 56% after reducing two convolutional layers to one which reduced the model complexity. This further emphasized the fact that simpler models will ultimately prone to achieve better results.

Initial Learning rate of 0.001 was varied up and down but the outcome did not show a huge improvement and hence they went back to keep the initial rate. They used 32 features per patch in their baseline model. Then increased it to 64 features per patch. It improved the accuracy by 1%. Then they kept increasing the features per patch until the improvement saturated at 256 features per patch. An improvement of 4% was gained by this variation which showed more features in CNN model leads for better accuracies. An increment from 2*2 receptive field setup to 4*4 does not improve the accuracy of the model.

Evaluated Weekly dataset achieved 55% overall accuracy rate. This did not matched or passed the Status Quo benchmark of 67% achieved by Logistic Regression Classifier. Overall CNN model performance was hampered by the inherited high complexity of the model.

4.4 Stacked Auto Encoders

Bao at el [12] has used SAE's in his study as a novel innovative idea in predicting financial time series. SAEs are used to exploit the hidden patterns of time series in an unsupervised manner. SAEs are built by stacking up single layer auto encoders where one auto encoder layer will connect to its predecessor and successor. One Auto-Encoder is trained at a single time to minimize the error propagation. Hence they manage to capture the invariant and abstract pattern in a timer series robustly.

Bao at el [12] used Stack Auto Encoders with Long Short-Term Memory to develop his model. There are 3 stages in the model.

(1) Wavelet Transform to eliminate noise associated with time series.

(2) Feeding data into Stacked Auto Encoders.

(3) Using LSTM Units to generate the final predictions.

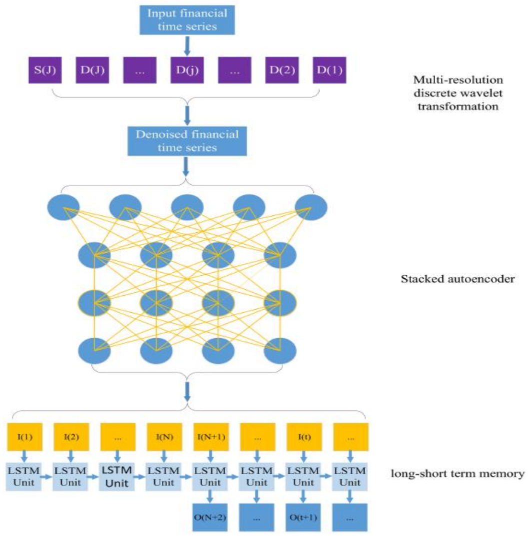


Figure 4.5: Deep Learning Architecture with Stack Auto Encoders

4.5 Long Short-Term Memory

Long Short-Term Memory is one variation which came from the concept of Recurrent Neural Network [13]. Use of RNN and LSTM for time series prediction is widely used in the field of data science. The RNN is a type of deep neural network architecture [14] that has a deep structure in the temporal dimension. It has been widely used in time series modelling. The assumption of a traditional neural network is that all units of the input vectors are independent of each other. As a result, the traditional neural network cannot make use of the sequential information. In contrast, the RNN model adds a hidden state that is generated by the sequential information of a time series, with the output dependent on the hidden state.

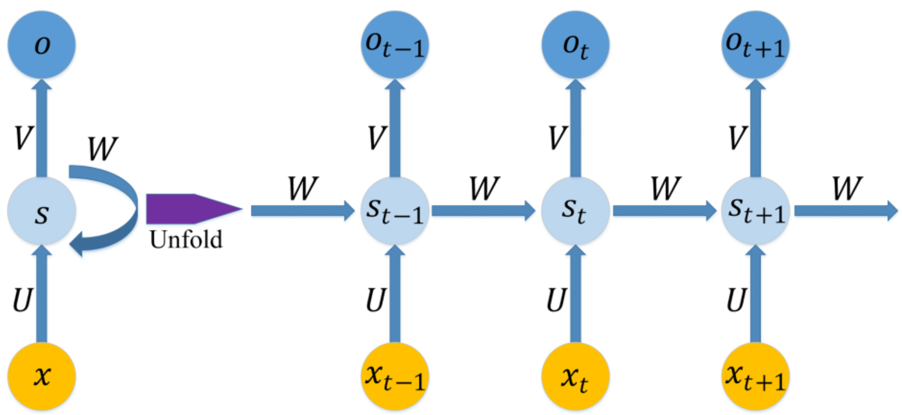


Figure 4.6: LSTM Model Architecture

The mathematical symbols in Figure 3 are as follows.

1. Xt is the input vector at time t.

2. St is the hidden state at time t; it is calculated based on the input vector and the previous hidden state. St is calculated by:

$$St = f\ (UXt + \ WSt)$$

Where f is the activate function, which has many alternatives such as sigmoid function and ReLU. The initial hidden state s0 for calculating the first hidden state s1 is typically initialized to zero.

3. Ot is the output at time t, which can be formulated as:

$$Ot = F(\,VSt)$$

4. U and V are the weights of the hidden layer and the output layer, respectively. W's are transition weights of the hidden state.

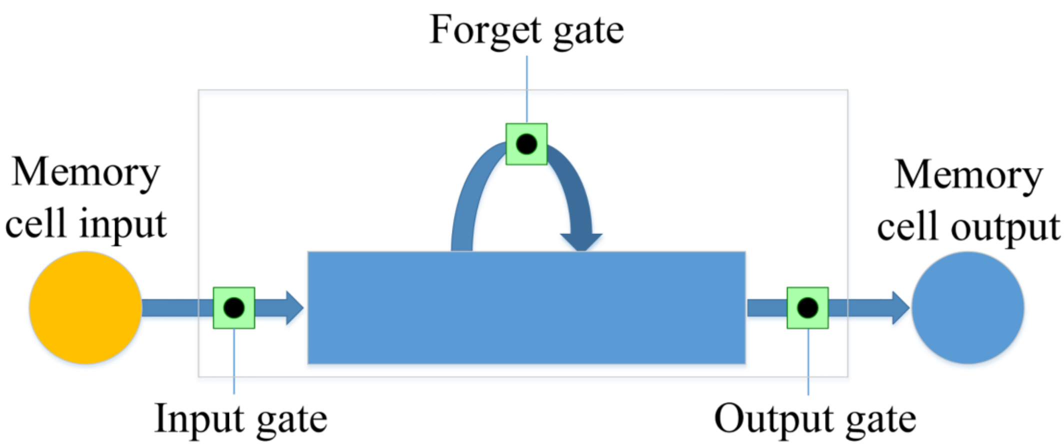


Figure 4.7: LSTM Cell with its Gates

Vanishing gradient problem is the Achilles heel in RNN models in modeling time series data [15]. LSTM is an effective mechanism for mitigating vanishing gradient issue. LSTM does it by using a concept called Memory Cells. Memory cell has four units: an input gate, an output gate, a forget gate and a self-recurrent neuron, which is illustrated in Figure 4.8. Gates determine the level of information transfer between neighboring cells. It also decided how much of information would be added to the memory cell as well. The amount of effect by the input signal to the memory states is controlled by the input gate. Output gate controls the level of effect which an output signal of a memory cell can generate to another cell or to itself [12].

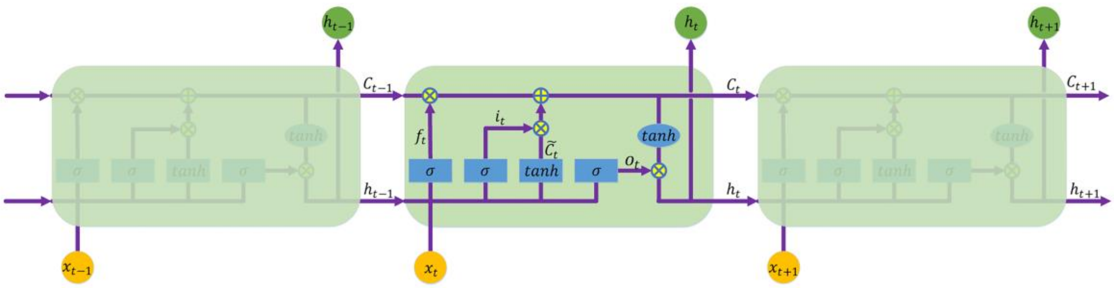


Figure 4.8: Repeating LSTM Modules

Figure 4.8 shows how LSTM gate fits in the neural network and how the values of each gate is passed through its gates. Mathematical notations of the above Figure 4.8 can be listed as below.

1. St is the input vector at time ‘t’.

2. Wi,Wf,Wc,Wo, Ui, Uf, Uc, Uo and Vo represent the weight vectors.

3. bi, bf, bc and bo represent the bias vectors associated with each cell.

4. ht indicates the value of memory cell at a given time ‘t’

5. it represents the value of input gate at time ‘t’.

6. ~ Ct represents the value of the candidate state at time t.

$$i_t = \sigma\ (W_i X_t + U_i * h_{t-1} + b_t)$$

$$\sim C_t = \tanh(W_c X_t + U_c * h_{t-1} + b_c)$$

7. ft is the value of the forget gate and Ct is the state of the memory cell at time t:

$$f_t = \sigma\ (W_f X_t + U_f * h_{t-1} + b_f)$$

$$C_t = i_t \sim C_t + \ f_t C_{t-1}$$

8. Ot represents the value of the output gate at time ‘t’.

9. ht represents the value of the memory cell at time ‘t’.

$$O_t = \sigma\ (W_o X_t + U_o * h_{t-1} + V_o C_t + b_o)$$

$$h_t = O_t * \tanh(C_t)$$

4.6 Gated Recurrent Units

Cho et al. [2014] introduced the Gated Recurrent Units. It was intended to adaptively capture the abstract relationships in time series data. GRU also controls the flow of information inside its unit by using gates. But GRU does not use separate memory cells to control the flow of data. [17]

Current activation h_t^j at a time t is a can be described using previous activation $h_{t-1}^j$and the candidate activation $\widehat{h_t^j}$:

$$h_t^j = (1 - z_t^j) h_{t-1}^j + z_t^j \tilde{h}_t^j,$$

Decision of how much the unit updates its activation, or content is decided by Update gate z_t^j in the above equation.

The update gate is computed by below equation:

$$z_t^j = \sigma \left(W_z \mathbf{x}_t + U_z \mathbf{h}_{t-1} \right)^j .$$

Linear sum is computed between new state and already existing state in the same way in LSTM. But GRU differentiates from LSTM as it does not control the level of exposure on the new state. GRU exposes the entire state without any control over exposure level.

The candidate activation $\widehat{h_t^j}$: can be computed as follows.

$$\tilde{h}_t^j = \tanh \left(W \mathbf{x}_t + U \left(\mathbf{r}_t \odot \mathbf{h}_{t-1} \right) \right)^j ,$$

When off (rjt close to 0), the reset gate effectively makes the unit act as if it is reading the first symbol of an input sequence, allowing it to forget the previously computed state.

The reset gate rjt is computed similarly to the update gate:

$$r_t^j = \sigma \left(W_r \mathbf{x}_t + U_r \mathbf{h}_{t-1} \right)^j .$$

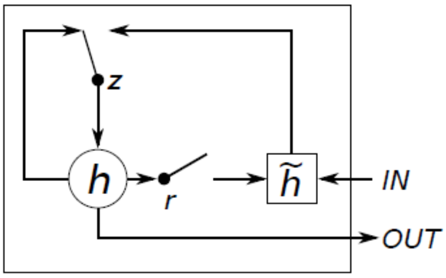


Figure 4.9: GRU Unit

4.7 Comparison between LSTM and GRU

LSTM and GRU both addresses the weakness of vanishing and exploding gradients in traditional recurrent neural networks. It also gives the opportunity to remember specific features in a stream over long period of time. Traditional RNN's usually have short term memory and loses or replaces valuable content as the time or series moves on.

LSTM unit used output gate as a checkpoint to determine the level of memory exposure to the next unit. But GRU's does not have such controlling gate hence it exposes its entire memory to the next unit or iteration. This will enhance the carry forward of past data in GRU and makes predictions more robust in problems which have high dependency on immediate past.

Location of the input gate and the reset gate are different in GRU. Memory Cell in LSTM computes its new state without any control over the information from the previous state. LSTM memory cell itself decides on how much newly computed memory should be added to its cell without the usage of a forget gate. GRU controls the information flow from the input when its new state is created but no control while the adding the new content to the new state [19]

4.8 Wavelet Based De-nosing

Wavelet Based De-noising is based on the idea that wavelet transform can achieve sparse representation of real world signals. Wavelet transform can concentrate the features of real signals in to a set of wavelet coefficients. Some Wavelet Coefficients are large and some are small. These small coefficients are knows as noise. So noisy coefficients can be removed by configuring a certain threshold. This does not effect to the quality of the original signal. Wavelet inverse transform could be used to reconstruct the original signal which is de-noised.

4.8.1 Fourier Transform

Fourier Transform decomposes a signal in time domain to infinite number of sine and cosine signals in frequency domain. These different frequency sine and cosine signals are known as harmonics.

For CTS	$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}\,dt$
For DTS	$F(\omega) = \sum_{-\infty}^{+\infty} f(n)e^{-j\omega n}$

We can use above two equations to do the Continuous and Discrete Domain Fourier Transforms. We are missing the time of an event occurrence when we do the transform from time to frequency domain.

4.8.2 Limitations of Fourier Transform

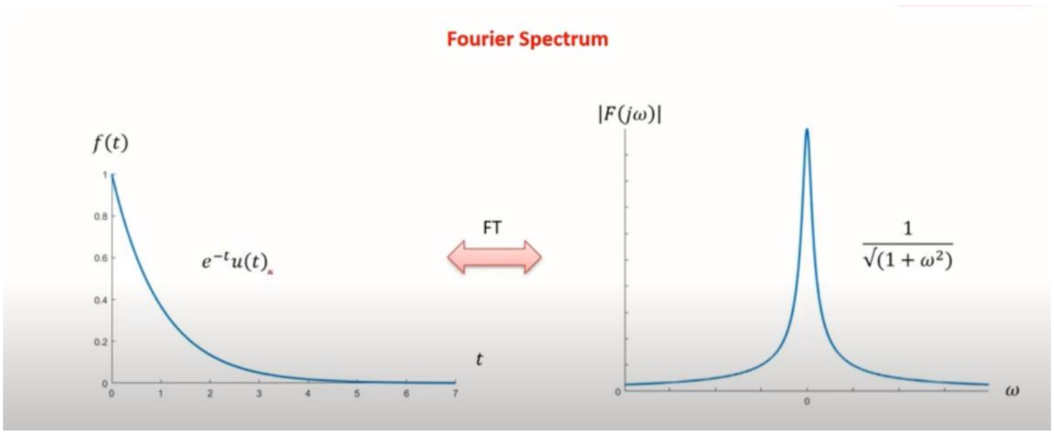


Figure 4.10: Fourier Spectrum

As you can see only the Frequency and Magnitude information will be available in Frequency domain losing its time information which was available in Time domain. If there is any discontinuity in the time domain we cannot see that in frequency domain.

Stationary Signal

Signals whose frequencies unchanged over time and all the frequencies exists all the time.

Non-stationary Signals.

Signals whose frequencies changed over time. “Chirp Signal” can be cited as an example.

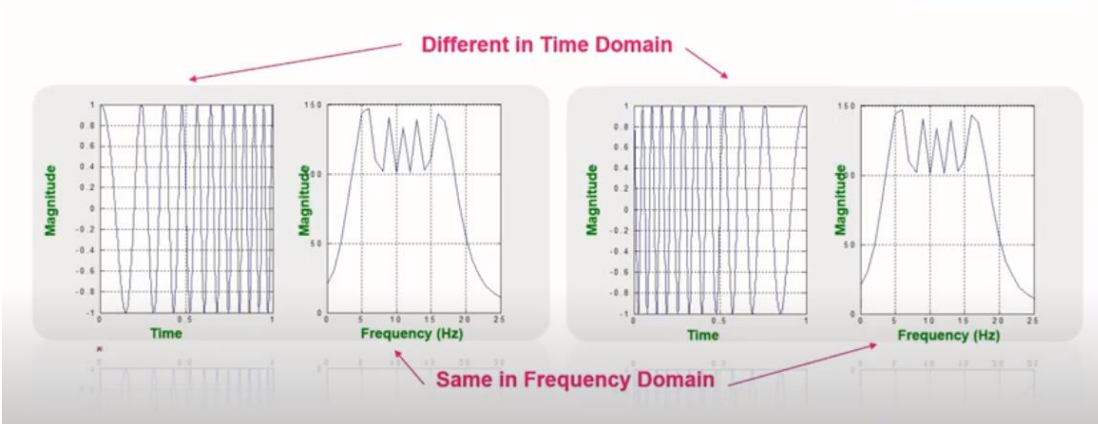


Figure 4.11: Time Domain Vs Frequency Domain

First signal in time domain is an increasing frequency signal and second time domain signal is a decreasing frequency signal. Both have same representation in frequency domain which is a limitation in Fourier transform.

4.8.3 Short-Time Fourier Transform

STFT is proposed to overcome the limitations of normal Fourier transform which causes loss of information in time domain. In this STFT method, a time window is used. The signal in that time window is considered as stationary and FT is applied. Then we get a 3-Dimensional chart with Time, Frequency and Magnitude as axes.

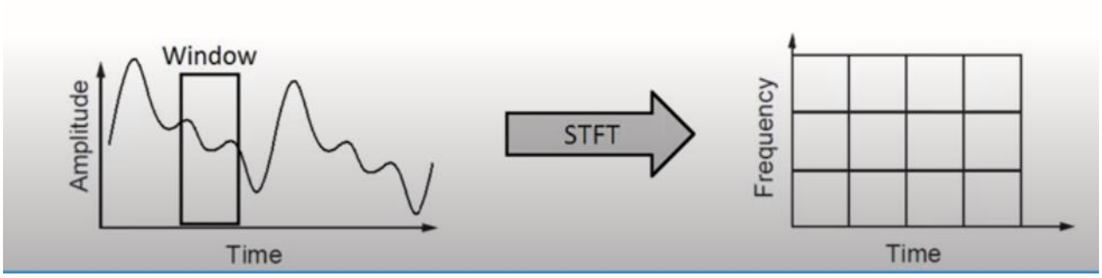


Figure 4.12: Short Term Fourier Transform Illustration

4.8.4 Drawbacks of Short-Time Fourier Transform

- Unchanged Window Size
- Dilemma of choosing Window Size
  - Narrow Window – Poor frequency resolution/ Good Time resolution
  - Wider Window – Good frequency resolution/ Poor Time resolution

4.8.5 Wavelets

A wavelet is a waveform effectively time limited duration which has an average value of zero. It is defined with the below equation.

$$\boxed{\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\,\psi\left(\frac{t-b}{a}\right) \quad a, b \in \mathbb{R}}$$

‘a’ is known as the dilation parameter where we can shrink or expand the wavelet by changing the value and ‘b’ is known as translation parameter which we can use to move the wavelet across time.

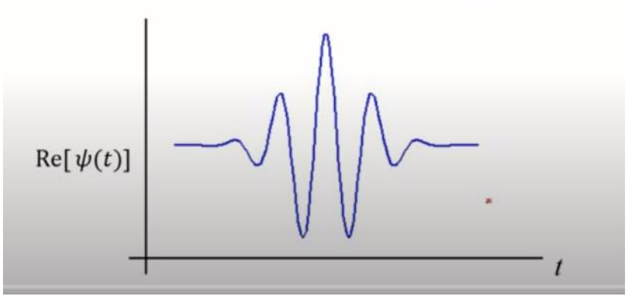


Figure 4.13: Example of a wavelet (Morlet’s Wavelet)

Continuous Wavelet Transform (CWT)

Integrated function when we change ‘a’ and ‘b’ continuously is known as Continuous Wavelet Transform. The equation of CWT for a function f(t) is given below.

$$\boxed{CWT(a,b) = \langle f, \psi_{a,b} \rangle = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} f(t).\, \psi^{*}\left(\frac{t-b}{a}\right) dt}$$

Wavelets with lower values of ‘a’ will filter out high frequencies in the signal and higher scales of ‘a’ will filter out lower frequencies in the signal. The resultant graph is known as scalogram. It is a representation between time, coefficients of ‘a’ and frequency. Drawback of CWT is obvious as it generates awful lot of data which is hard to handle. So it paves the way for Discrete Wavelet Transform (DWT).

Discrete Wavelet Transform (DWT)

Values for ‘a’ and ‘b’ parameters which results in discrete wavelet transform. DWT usually takes 2 forms depending on the selection of values for ‘a’ and ‘b’.

1. Redundant Wavelet Transform.

‘a’ is chosen as the integer powers of fixed dilation parameter. (a_0 >1)

$$a = a_0^m$$

Different values of ‘m’ corresponds to wavelets of different widths. Narrower wavelets are translated by small steps and wider ones by larger steps. Therefore ‘b’ translation parameter is chosen as below.

$$b = nb_0 a_0^m$$

Where $b_0 > 0$

2. Multi Resolution Analysis.

This is the most famous method out of two and introduced by S.Mallat in 1988. He proposed to choose scales and positions based on powers of two which is also called in mathematics as dyadic scales and positions. Then analysis becomes much simpler as well as accurate.

For some special choice of wavelet function and choice of a0 and b0 will constitute an orthonormal basis for L2.

a_0 =2 b0=1 and below wavelet function makes such orthonormal bases for L^2.

4.8.6 Analysis and Synthesis

Analysis

Decomposition of the signal to high frequencies and low frequencies using high pass and low pass filters.

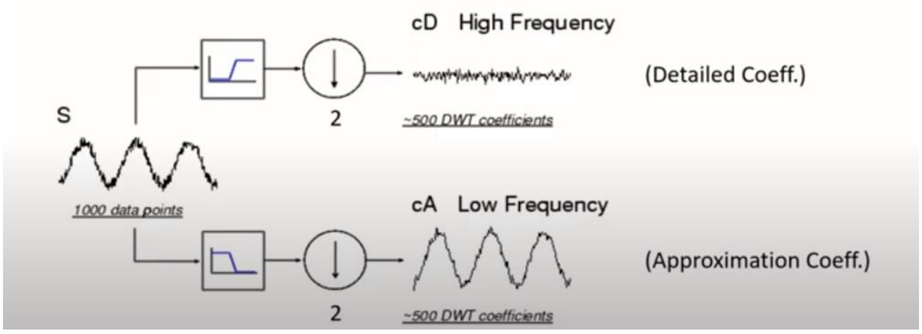


Figure 4.14 Signal Decomposition using High pass and Low Pass Filters

As indicated above signal will be passed through a high pass filter and down sampled by two then we get detailed coefficients of wavelets. When the same signal is passed via Low pass filter and down sampled then it gives approximation coefficients. It contains low frequencies. These filters are specially designed filters rather than normal LPF or HPF which gives wavelet coefficients when signal is convolved with the impulse response of the filter function.

Synthesis

It is the inverse process of analysis where we reconstruct the original signal using detailed and approximate coefficients.

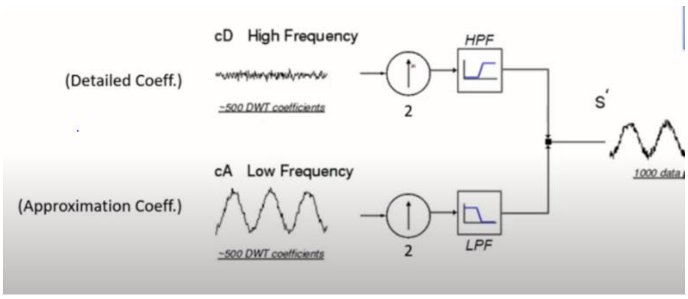


Figure 4.15: Signal Synthesis using High pass and Low Pass Filters

Multilevel Decomposition

We can use the approximate coefficients in the analysis process and perform analysis again and again. This is called multilevel decomposition and removes noisy features in the signal.

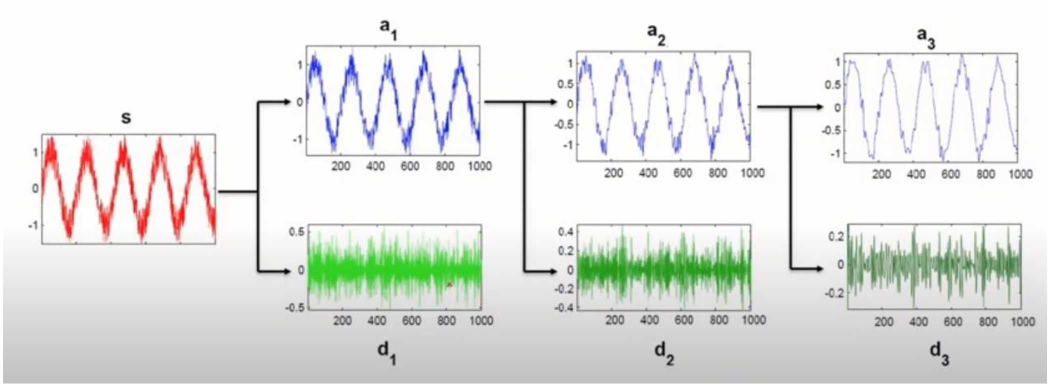


Figure 4.16: Multi Level Signal Synthesis using Filters

4.8.7 Types and families of Wavelets

Wavelets types commonly used are listed below.

1. Haar
2. Symlets
3. Mayer
4. Daubechies
5. Coiflets

There are two broad wavelet categories of depending on their properties.

1. Wavelet with filters
    a. With compact support.
        i. Orthogonal
            1. DB/Haar/Sym/Coif
        ii. Bi-Orthogonal
            1. Bior.
    b. Without compact support.
        i. Orthogonal

1. Meyr, Dmey

2. Wavelet without filters.
    a. Real
        i. Gaus/Mexh/Morl
    b. Complex
        i. Cgau/shan/fbsp

4.8.8 Applications of Wavelets

- Signal Compression
- Signal Denoising

Signal denoising is done by applying a threshold to the coefficients of the detailed coefficient output.

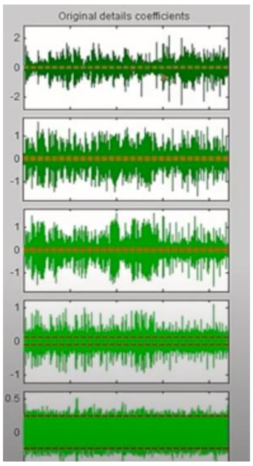


Figure 4.17: Applying a threshold to remove high frequencies

As shown in the figure we can change the threshold and remove unnecessary high frequency parts in the signal.

- Discontinuity Detection
- Image Enhancement.

5. PROPOSED METHOD

5.1 Data Set

Tadawul Exchange Data which is owned by DirectFN Private Limited were used for the research. Most of the data are stored as structured data in relational databases. For other data which is not available with the DirectFN Private Limited was obtained via publicly available credible data sources.

Table 5.1: Data Sources for the research

OHLC Data of TASI Index	Tadawul Stock Exchange data feed Via DirectFN
Technical Indicators	Calculated based on OHLC data
Exchange Rates	Publicly Available External Data Source
Any Data related to index constituents of TASI	Tadawul Stock Exchange data feed Via DirectFN

Training, Validation and Testing Data Sets will be created from the above data sources. Training, Validation and Testing Data Sizes are as follows.

Table 5.2: Training, Validation and Testing Set Description

Data Set	**Period**	**Number of records (Apprx)**
Training Set	22 years	6408 Records
Validation Set	2 years	500 Records
Testing Set	2 years	500 Records

Research is extended to identify the success as well as behavior of individual stocks when LSTM, GRU and Hybrid of LSTM and GRU methods are applied for future close price prediction. I have choose five most active stock symbols from five different sectors to analyze with my deep neural network.

Table 5.3 Sectors and No of Records of chosen Stocks

Symbol	**Years**	**Number of Records**	**Sector**
4050 (Saudi Automotive Services Corporation)	29 Years	7718 records	Retail Sector
4040 (Saudi Arabian Public Transport Company)	29 Years	7752 records	Transport Sector
2120 (Saudi Advanced Industries Corporation)	26 Years	6123 records	Diversified Financials
2020 (Saudi Agri Nutrients Corporation)	29 Years	7614 records	Materials Sector
1150 (Alinma Bank – Saudi Arabia)	14 Years	3407 records	Bank Sector

5.2 Data Preprocessing

OHLC and other technical indicators will be preprocessed to remove or address anomalies and missing data points. Then time series data will be normalized via min-max normalization which will make the data range between 0 and 1 which improves accuracy in deep learning models. Equation (2) states the min-max transformation.

$$v_i' = \frac{v_i - v_{min}}{v_{max} - v_{min}} \qquad (2)$$

where v = (v1, v2, . . . , vn), v_i' is the i'th normalized data.

Zero-mean was used for further data processing so each feature has a mean value of zero. It is advantageous to center the data for practical reasons. The formula is as follows:

$$v_i'' = v_i' - v_{mean}'$$

Where v_i'' is the zero mean data.

5.3 Principal Component Analysis

Principal component analysis was used as the strategy to reduce the dimensionality of the input feature set and extract the best features which has an impact on the index price of the TASI Index. PCA was used to build two out of four models. One model used PCA for both OHLC and Technical Indicator data and the other model only used PCA for Technical Indicator data.

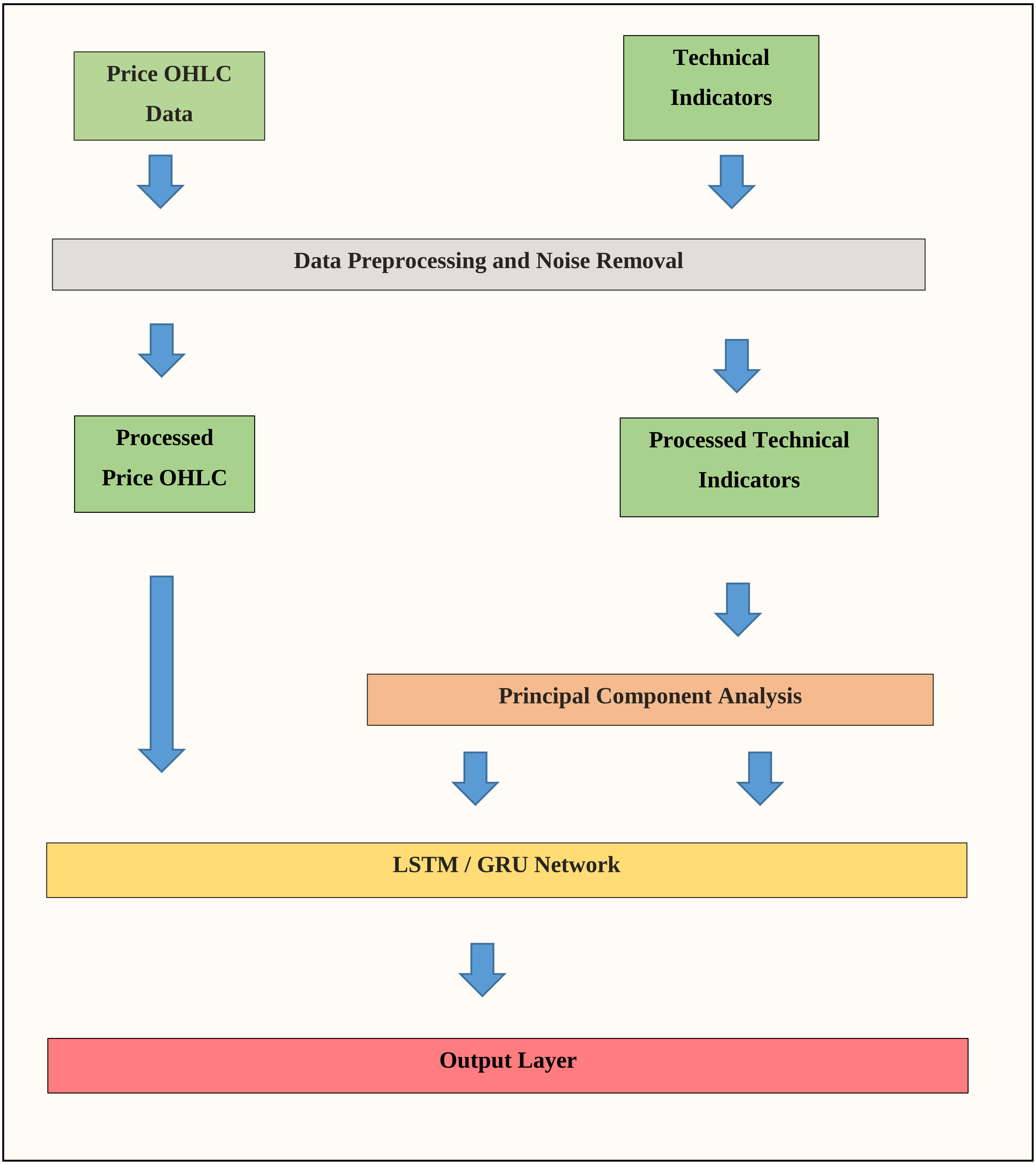
5.4 LSTM and GRU based Prediction

The features extracted from PCA will be fed into a prediction network where it will use the past data points as sequential data series to train the model. Both LSTM and GRU models individually and in a hybrid mode were used for prediction task. All LSTM, All GRU and Hybrid models were compared in terms of computational efficiencies and performances.

Effectiveness of LSTM/GRU architectures primarily decides by the availability of training and validation data. Apart from that Neuron Count in each layer as well as total number of layers act as key parameters in the network and there are no specific set of rules on how to select them. It is basically done via trial and error method.

5.5 Proposed Architecture

Figure 5.1 shows the diagram of the proposed architecture where raw OHLC data will be directly fed to the Depp Neural Network. Technical Indicators which will be calculated by using OHLC and volume data will be fed to the PCA layer in order to reduce the dimensionality. This will be main model evaluated in the research but three other models will be created by altering the data flow paths to the deep neural network. There will be 4 models evaluated in the research which will be explained in the next section.



5.6 Different Model Figure 5.1: Proposed Architecture

5.6.1 Model – A

Figure 5.2 shows the data flow of the Model A which use OHLC data only for its model creation. The OHLC data will be directly fed to the deep neural network without going through a PCA Layer.

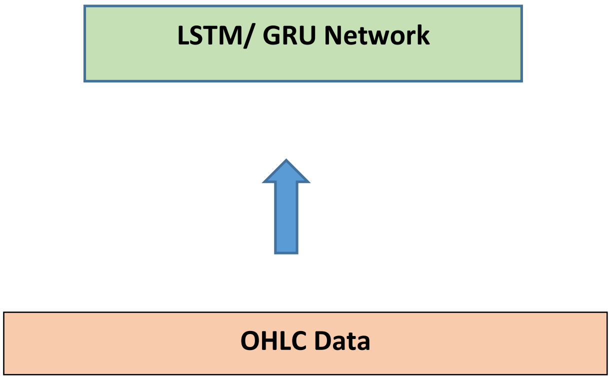


Figure 5.2: Proposed Model A

5.6.2 Model – B

Figure 5.3 shows the data path of the second model which is designed to feed the raw OHLC price data and technical indicators directly to the PCA layer and the entire output will be fed to the LSTM or GRU layer.

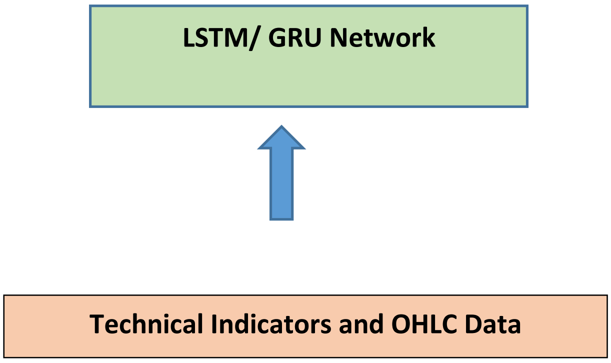


Figure 5.3: Proposed Model B

5.6.3 Model – C

Figure 5.4 illustrates the data path of the third model which is designed to feed the raw OHLC price data and Technical Indicators to the PCA layer and then the output will be fed to the LSTM/GRU layer.

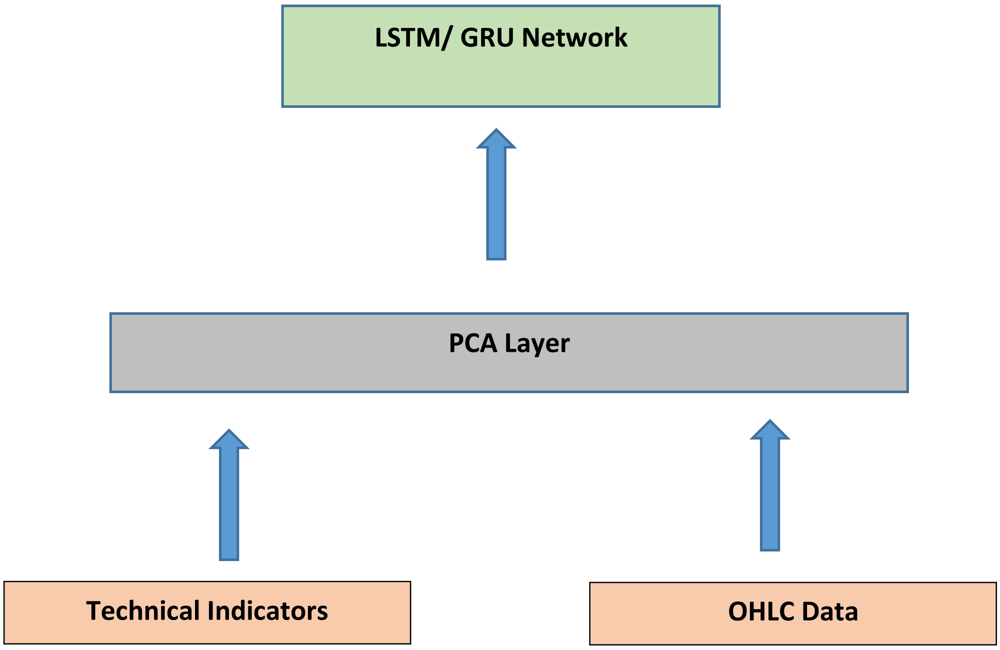


Figure 5.4: Proposed Model C

5.6.4 Model – D

Figure 5.5 illustrates the data path of the fourth model which is designed to feed Technical Indicator data to the PCA layer and then the output will be fed to the LSTM/GRU layer. Raw OHLC price data will be directly to the to the LSTM or GRU layer.

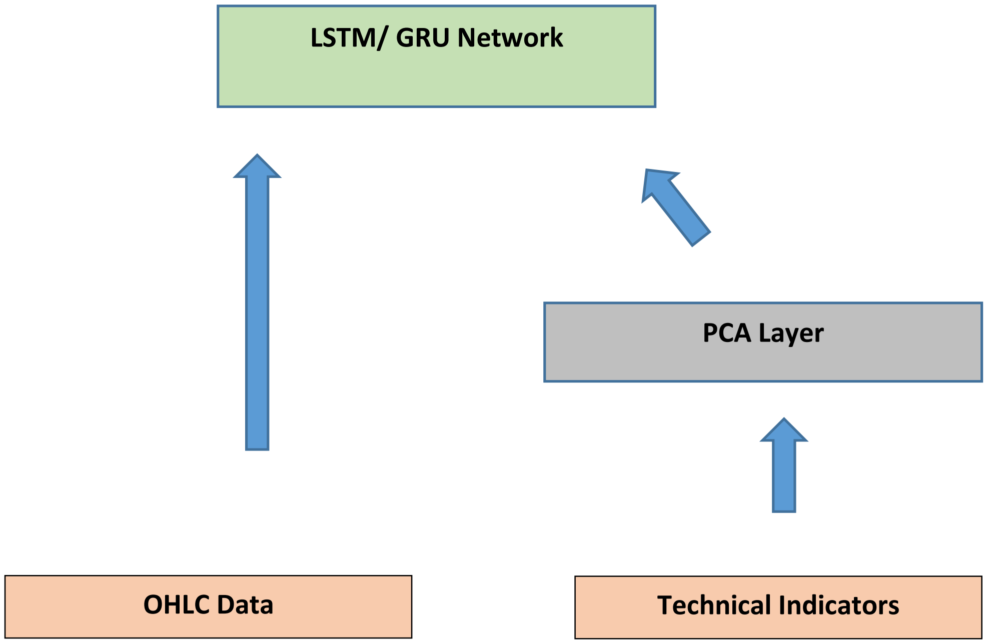


Figure 5.5: Proposed Model D

5.7 Technical Indicators which will be used with the model

5.7.1 Accumulated Distribution (AD)

The accumulation/distribution indicator (A/D) is a cumulative indicator which assess whether a stock is being accumulated or distributed by using the stock price and volume.

Money Flow Multiplier = {(High −Low) - (Close Price−Low)}/ (High−Close Price)

Where:

Low=Lowest price of the intended period

High= Highest price of the intended period

Money Flow Volume= Money Flow Multiplier ×Period Volume

Accumulated Distribution =Previous Accumulated Distribution + Current Period Money Flow Volume

5.7.2 Moving Average Convergence Divergence (MACD)

MACD reveals the correlation between two moving averages of stock price. It also elaborates the direction and duration of the relationship as well.

Moving average convergence divergence (MACD) is a trend-following momentum indicator that shows the relationship between two moving averages of a security's price. The MACD is calculated by subtracting the 26-period exponential moving average (EMA) from the 12-period EMA. Equation (3) states the above mentioned formula.

$$\mathrm{MACD} = \mathrm{EMA}(CL, 12) - \mathrm{EMA}(CL, 26) \qquad (3)$$

5.7.3 Chaikin Oscillator

Chaikin Oscillator is calculated by first calculating 3-day and 10-day exponential moving average (EMA) of accumulation-distribution line and then subtracting 10-day EMA from 3-day EMA. This indicator measures the momentum of oscillations around the accumulation-distribution line. Equation (4) states the formula in calculating Chaiking Oscillator.

$$\mathrm{CHO} = \mathrm{EMA}(AD, 3) - \mathrm{EMA}(AD, 10) \qquad (4)$$

5.7.4 Highest and lowest of time period t

Highest (t) is the highest closing price value during the past t trading days. Lowest (t) is the lowest closing price value within the past t trading days. Equation (5) states the formula used to calculate the above mentioned indicator.

$$Lowest(t) = \min(CL_i) \qquad Highest(t) = \max(CL_i) \qquad (5)$$

5.7.5 Stochastic Oscillator

Stochastic Oscillator is a technical indicator which is used to measure the momentum of a stock. It compares the closing price of a certain stock to a range of its own prices in a definitive time period. The sensitivity of the oscillator can be altered by tuning the number of time periods. Usually the values lie between 0 to 100 ranges.

$$\text{SO-\%K} = (\text{CL}-\text{Lowest }(5))/\text{Highest }(5) - \text{Lowest }(5) \times 100\% \qquad (6)$$

$$SO - \%D = MA(STOS - \%K, 3) \qquad (7)$$

5.7.6 Volume Price Trend Indicator

The Volume Price Trend Indicator (VPT) is a stock market indicator that helps traders relate a stock's price and trading volume. It helps in identifying the parity between the supply and demand for a stock and also helps in predicting the price of a stock, both in direction and magnitude

$$VPT = VPT_{previous\ day} + VO * \frac{CL - CL_{previous-day}}{CL_{previous-day}} \qquad (9)$$

5.7.7 William's R Indicator

Williams Percent Range, is a type of momentum indicator that ranges from 0-100. It is used to measure the behavior of todays close price of a stock to its own close prices over an 'n' period. Usually period is taken as 14 days when calculating William's %R.

$$\text{W-R\%} = \frac{\text{Highest}(n) - CL}{\text{Highest}(n) - \text{Lowest}(n)} \times 100\%$$

5.7.8 Relative Strength Index (RSI)

RSI is a famous technical indicator used in technical analysis by investors. Stochastic formula is applied to a group of RS values instead of price data. The output gives an indication whether these RS values are overbought or oversold. Usually RSI ranges between 0 and 1.

$$\text{RSI} = 100 - \frac{100}{1 + \text{RS}}$$

5.7.9 Momentum Indicator (MOME)

Short term price swings is used to measure the magnitude of the price momentum. Price pivots mark the starting and end of each swing. Steep slope combined with long price swing indicates a strong momentum while shallow slope combined with short price swing indicates weak momentum.

Length of the upswings indicate the levels of momentum. Long upswings indicate strong increasing momentum while short swings suggest weak declining momentum. Equal length swings indicate that momentum remains the same.

$$\mathrm{MOME}(n) = CL_t - CL_{t-n}$$

5.7.10 Volume / Price Rate of Change (VROC and PROC)

Rate of Change based on price is known as PROC and rate of change based on Volume is known as VROC. Both are momentum based indicators. It measures the percentage change in price or volume between current values and values of certain 'n' periods ago.

5.7.11 On Balance Volume (OBV)

OBV is a technical momentum indicator which is used to predict the market changes depending on trade volume changes. OBV assumes volume as the main force behind market movements.

$$\text{If } CL \geq CL_{\text{previous-day}},\ \mathrm{OBV} = \mathrm{OBV}_{\text{previous-day}} + VO$$
$$\text{If } CL < CL_{\text{previous-day}},\ \mathrm{OBV} = \mathrm{OBV}_{\text{previous-day}} - VO$$

5.7.12 Average Directional Index (ADX)

ADX is used to quantify trend strength. ADX calculations are based on a moving average of price range expansion over a given period of time. The default setting is 14 bars, although other time periods can be used. ADX can be used on any trading vehicle such as stocks, mutual funds, exchange-traded funds and futures.

6. RESULTS AND ANALYSIS

This Chapter states the overall results of the different models and gives a comparison between accuracies of the models. It also evaluates and states the accuracy levels of traditional statistic models and other machine learning models. It will help to gauge the performance of the new hybrid models with the already established methods.

6.1. Benchmark Model Prediction Results

Table 6.1 shows the list of basic statistical models and other prevalent machine learning models which is used for comparison purposes. It will give a clear baseline and will be able to gauge the improvement added by the LSTM, GRU and LSTM plus GRU Hybrid models.

Table 6.1 Description of Benchmark Models

Model Name	**Description**
Moving Average Model	Statistical Method. Rolling Window size of 3 days is used.
Exponential Moving Average Model	Statistical Method. Rolling Window size of 3 days is used.
ARMA Model	Statistical Method. Use Auto Regression and Moving Average to express a time series.
GARCH Model	Statistical Method. Integrates Variance of Error with Auto Regression to express a time series.
Support Vector Regression – OHLC only	Machine Learning method. Only used OHLC values of the index/stock to create the model.

Support Vector Regression – OHLC and Technical Indictors	Machine Learning method. Used OHLC values and Technical Indicators of the index/stock to create the model.
Feed Forward Neural Network (FFNN)	Normal Feed Forward Neural Network with depth of 5.

6.2 Short Term Evaluation and Analysis

Short Term evaluation is defined as predicting the close price behavior for next 25 day period. It is primarily important for retail customers who anticipate returns in the short run. It also has the highest impact from daily socio-political behaviors of the market and its companies making it the hardest to predict using micro-economic indicators.

6.2.1 Tadawul All Share Index (TASI) Evaluations

Prediction was done for Tadawul All Share Index commonly known as TASI. Results of the benchmark models for the Short Term are as follows.

Table 6.2 TASI Index – Short Term Results of the Bench Mark Models

Model	**MSE**	**RMSE**	**MAE**	**MAPE**
Moving Average	0.0000423072	0.0065043995	0.0056059743	0.0117649364
Exponential Moving Average	**0.0000379175**	**0.0061577196**	**0.0052028854**	**0.0109133223**
ARMA	0.0000823100	0.0286897194	0.0060288737	0.0299129834
GARCH	1.4007207487	1.1835204893	0.9830466942	28.2985343868
SVM	0.0001862125	0.0136459690	0.0134266932	0.0283066540
SVM	0.0022269817	0.0471909065	0.0470076420	0.1094335888
FFNN	0.0001496288	0.0122322850	0.0108796209	5.1189761162

Exponential Moving Average which is a statistical method recorded the best overall Mean Absolute Error Percentage value in benchmark models. A variable moving average window was tried and best results were obtained for rolling window size of three.

Window Size

Window Size is the size of the moving window which is used to predict the next prediction. A Windows Size of ‘n’ mean that algorithm use previous ‘n’ values for predicting ‘n+1’th value. Window size imply how many prior days will be used for predicting the next day close price in this research scope.

6.2.1.1 Model A Evaluations

6.2.1.1.1 Optimizing the Window Size

All other parameters were kept a constant value and observed the behavior of the Mean Squared Error with the variation of Window Size to obtain a precise window size in days. The experiment is performed 10 times and Mean Squared Error is averaged out at the end. Figure 6.1 shows the variation of MSE with the variation of Window Size. It shows that a moving window size of 1 is giving the best accurate results. It hints that tomorrow's close price only depends mostly on yesterday's close price.

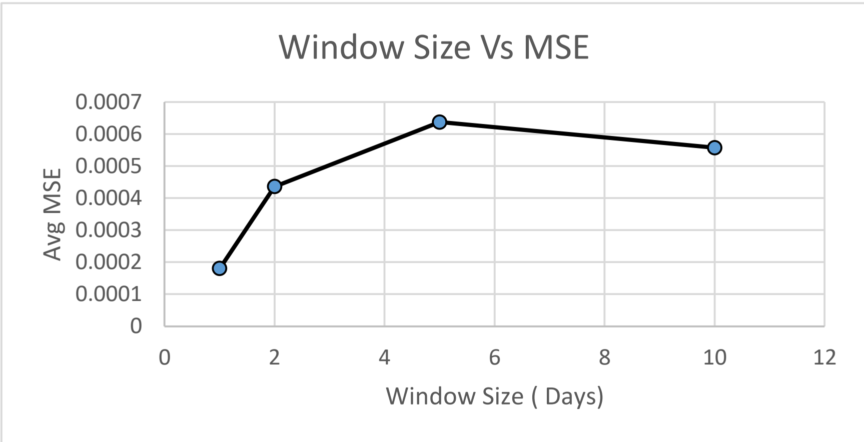


Figure 6.1: Model A –Short Term Window Size Vs. MSE Variation

6.2.1.1.2 Optimizing the Neuron Count in Each Layer

Neuron Count in each layer was also tuned by keeping all other parameters constant and running the test 10 separate times to record the average performance. Figure 6.2 show that best accuracies are obtained when the neuron count per layer is between 20 and 30 for the model which feeds OHLC data directly to the Neural Network (Model A).

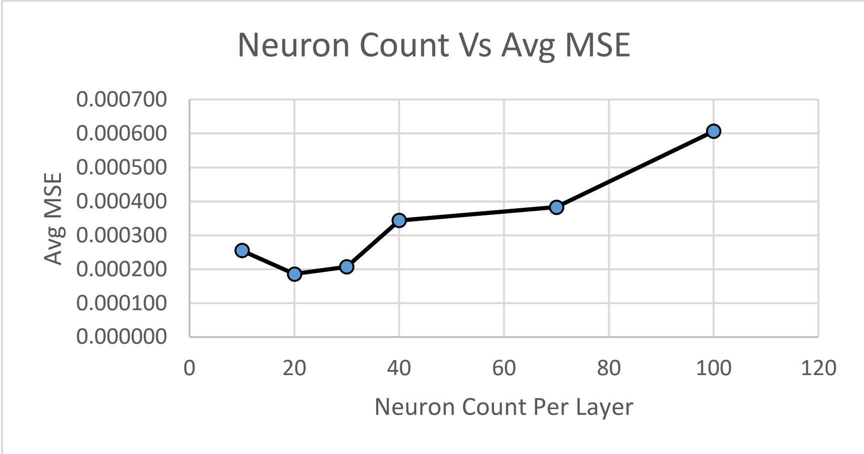


Figure 6.2: Model A –Short Term Neuron Count Vs. MSE Variation

6.2.1.1.3 Optimizing the training Batch Size of the Network

Optimum Batch Size was found by keeping all other parameters in a constant value. Batch Size search was done following a grid search methodology. Each batch size point was evaluated repeatedly 10 times to obtain the averaged Mean Squared Error. Figure 6.3 show the batch size of 16 gave the best results for the model which feeds OHLC data directly to the Neural Network (Model A).

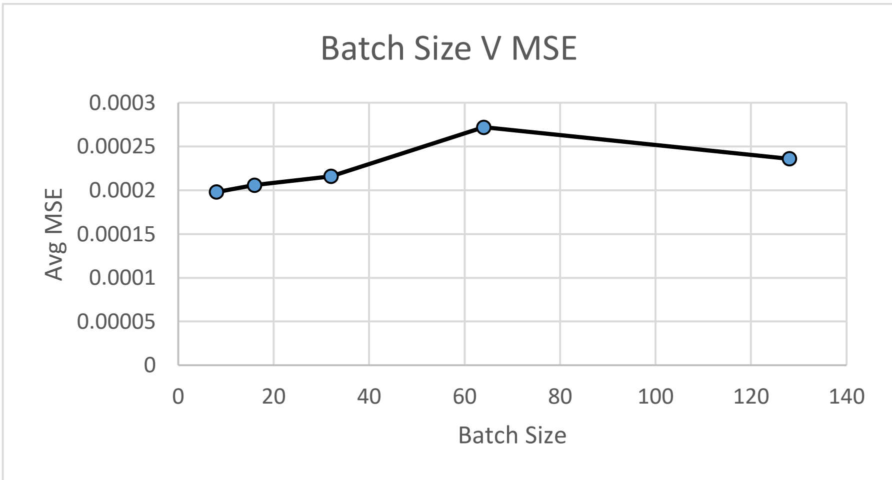


Figure 6.3: Model A –Short Term Batch Size Vs. MSE Variation

6.2.1.1.4 Deep Neural Network Layer Count Tuning

Model which feeds OHLC data directly to the Neural Network (Model A) is trained and tested with All LSTM layers, All GRU layers and Hybrid of LSTM and GRU layers. Figure 6.4 and Figure 6.5 shows the variation of Mean Squared Error with No of LSTM and GRU Layer count. Best Performance for both LSTM and GRU networks were given when the layer count is between 1 and 5.

LSTM Layer Count versus Mean Squared Error Variation

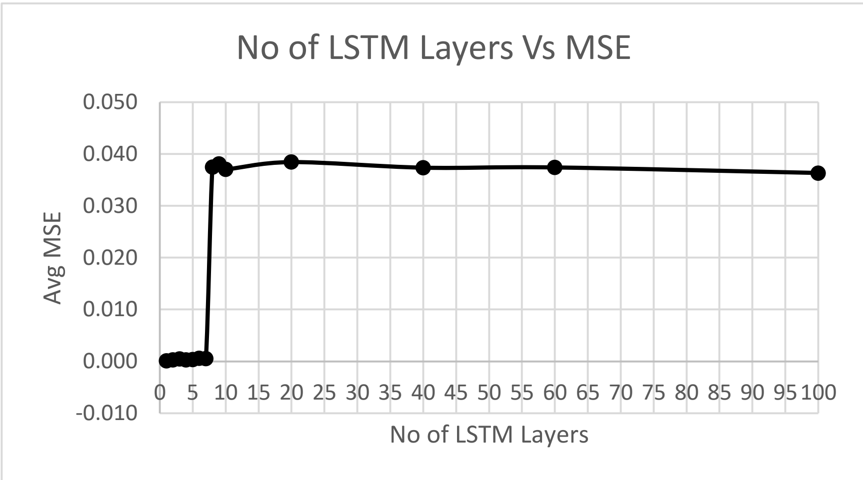


Figure 6.4: Model A –Short Term LSTM Layer Count Vs. MSE Variation

GRU Layer Count versus Mean Squared Error Variation

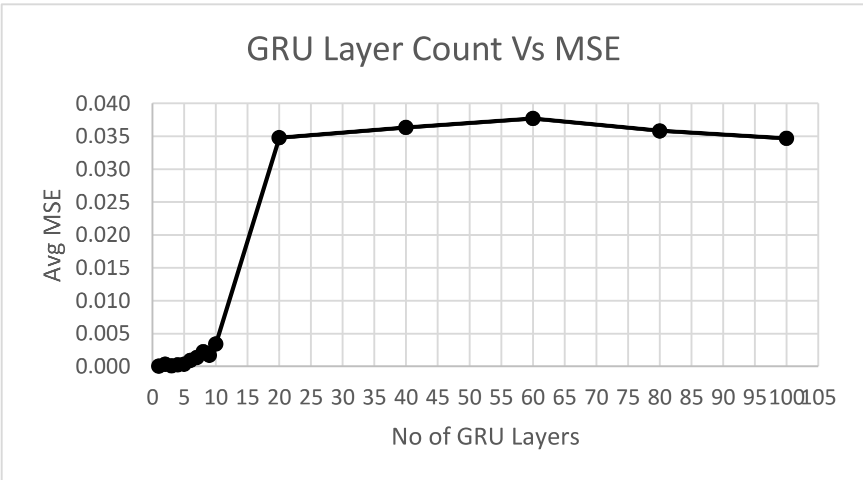


Figure 6.5: Model A –Short Term GRU Layer Count Vs. MSE Variation

6.2.1.1.5 Overall Model Performance

Model which feeds OHLC data directly to the Neural Network (Model A) is trained with all LSTM, All GRU and Hybrid network with LSTM and GRU Layers. Table 6.3 tabulates the best results of each network. Hybrid Network which comprised of LSTM and GRU Layers are giving best performance in terms of Maximum Absolute Percentage Error in Model A. MAPE of 1.3% is considered as good and reliable as it is closer to 1% benchmark. Figure 6.6 shows how Actual Close price varies with predicted close price over a period of 25 days.

Table 6.3 TASI Index – Short Term Results of Model A

Model	**Mean Squared Error**	**Root Mean Squared Error**	**Mean Absolute Error**	**Mean Absolute Percentage Error**
LSTM	0.0000357490	0.0059790504	0.0045558601	1.8579021692
GRU	0.0000343544	0.0058612647	0.0044072508	1.7966028452
LSTM + GRU	**0.0000192104**	**0.0043829624**	**0.0032739572**	**1.3478728533**

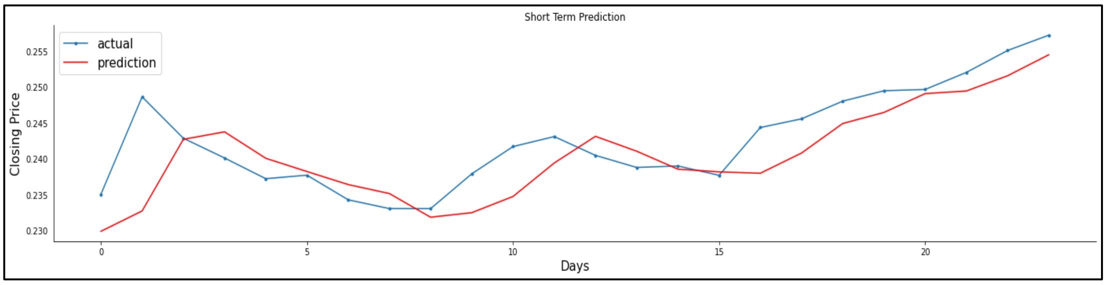


Figure 6.6: Model A Short Term – TASI Actual Close Price vs. Predicted Close

6.2.1.2 Model B Evaluations

6.2.1.2.1 Optimizing the Window Size

All other parameters were kept a constant value and observed the behavior of the Mean Squared Error with the variation of Window Size to obtain a precise window size in days. The experiment is performed 10 times and Mean Squared Error is averaged out at the end. Figure 6.7 shows the variation of MSE with the variation of Window Size. It shows that a moving window size of 1 is giving the best accurate results. It hints that tomorrow's close price only depends mostly on yesterday's close price.

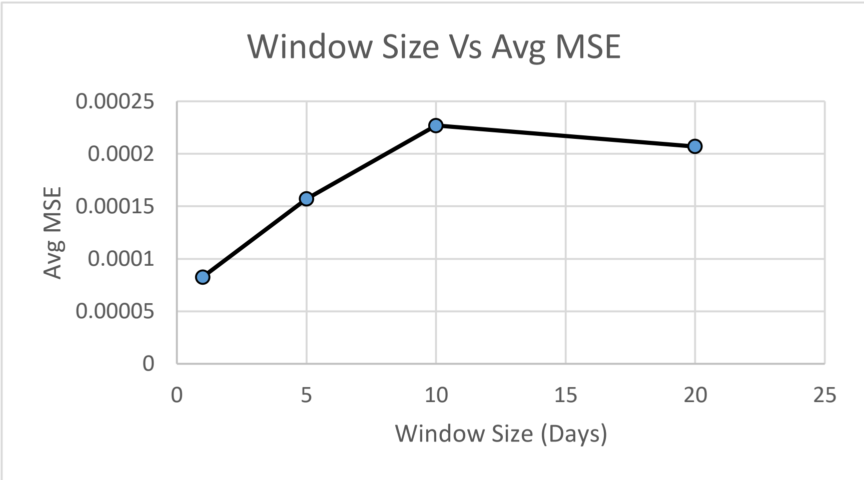


Figure 6.7: Model B –Short Term Window Size vs. MSE Variation

6.2.1.2.2 Optimizing the Neuron Count in Each Layer

Neuron Count in each layer was also tuned by keeping all other parameters constant and running the test 10 separate times to record the average performance. Figure 6.8 show that best accuracies are obtained when the neuron count per layer is between 20 and 40 for the model which feeds OHLC and Technical Indicator data directly to the Neural Network (Model B)

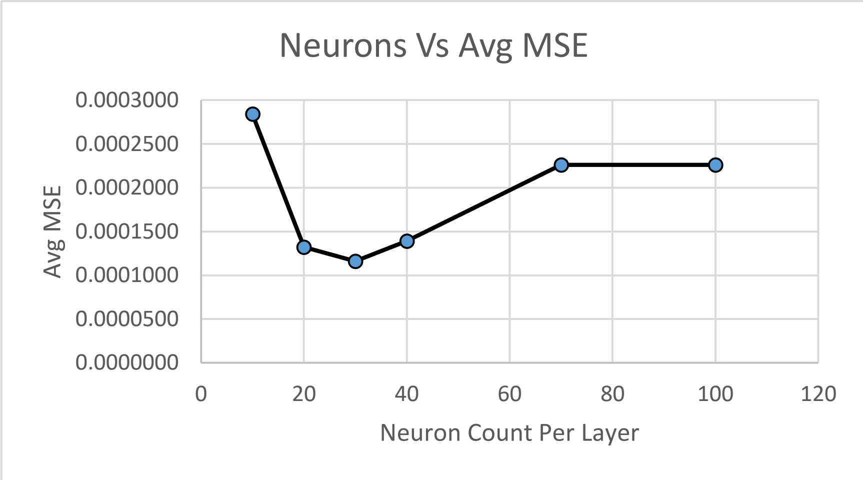


Figure 6.8: Model B –Short Term Neuron Count vs. MSE Variation

6.2.1.2.3 Optimizing the training Batch Size of the Network

Optimum Batch Size was found by keeping all other parameters in a constant value. Batch Size search was done following a grid search methodology. Each batch size point was evaluated repeatedly 10 times to obtain the averaged Mean Squared Error. Figure 6.9 show the batch size of 16 and 32 gave the best results for the model which feeds OHLC and Technical Indicator data directly to the Neural Network (Model B).

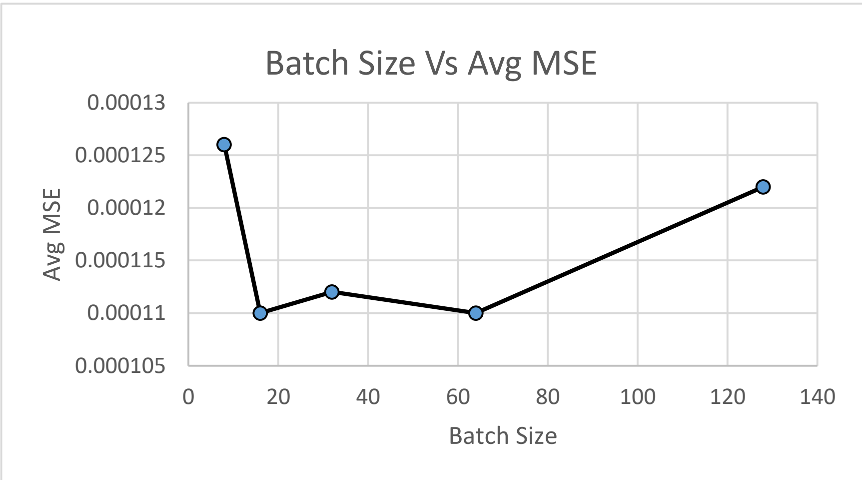


Figure 6.9: Model B –Short Term Batch Size vs. MSE Variation

6.2.1.2.4 Deep Neural Network Layer Count Tuning

Model which feeds OHLC data and Technical Indicator data directly to the Neural Network (Model B) is trained and tested with All LSTM layers, All GRU layers and Hybrid of LSTM and GRU layers. Figure 6.10 and Figure 6.11 shows the variation of Mean Squared Error with No of LSTM and GRU Layer count. Best Performance for both LSTM and GRU networks were given when the layer count is between 1 and 4.

LSTM Layer Count versus Mean Squared Error Variation

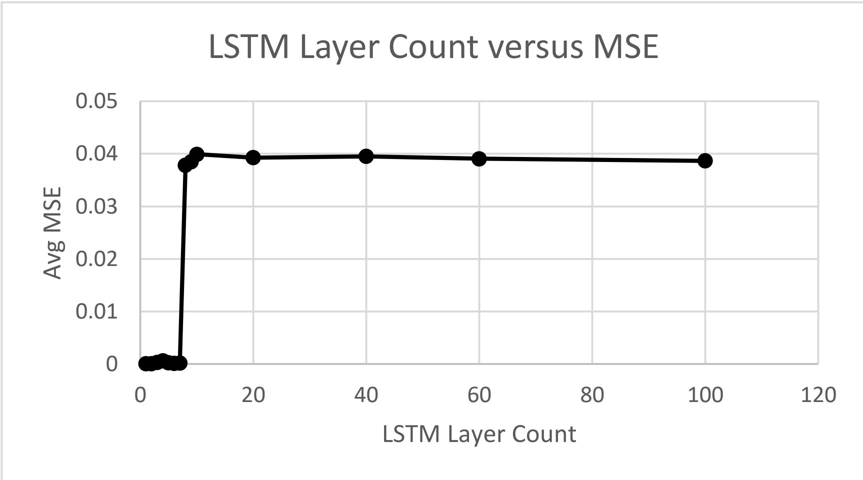


Figure 6.10: Model B –Short Term LSTM Layer Count vs. MSE Variation

GRU Layer Count versus Mean Squared Error Variation

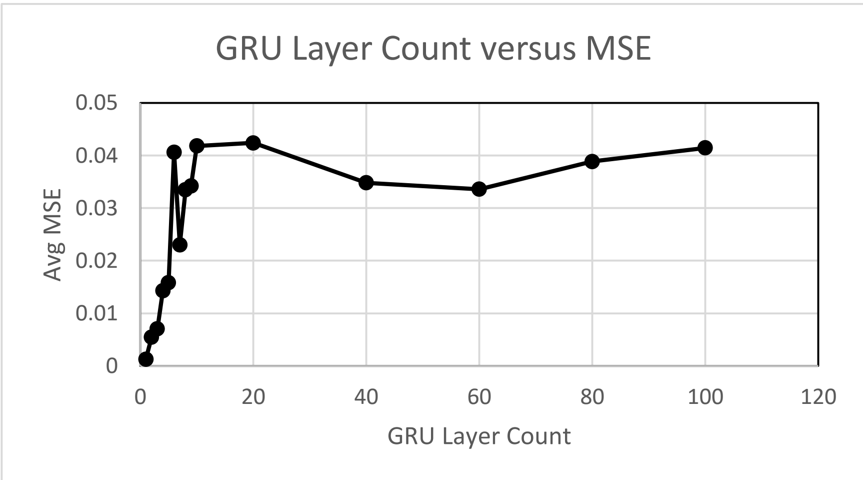


Figure 6.11: Model B –Short Term GRU Layer Count vs. MSE Variation

Best Performance for both LSTM and GRU layers were given when the layer count is between 1 and 5.

6.2.1.2.5 Overall Model Performance

Model which feeds OHLC and Technical Indicator data directly to the Neural Network (Model B) is trained with all LSTM, All GRU and Hybrid network with LSTM and GRU Layers. Table 6.4 tabulates the best results of each network. Hybrid Network which comprised of LSTM and GRU Layers are giving best performance in terms of Maximum Absolute Percentage Error in the Model B.

Table 6.4 TASI Index – Short Term Results of Model B

Model	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	0.0000195695	0.0044237440	0.0033501536	1.3726196289
GRU	0.0000197540	0.0044445479	0.0035526988	1.4671877623
LSTM + GRU	**0.0000232917**	**0.0048261494**	**0.0033511459**	**1.3699408579**

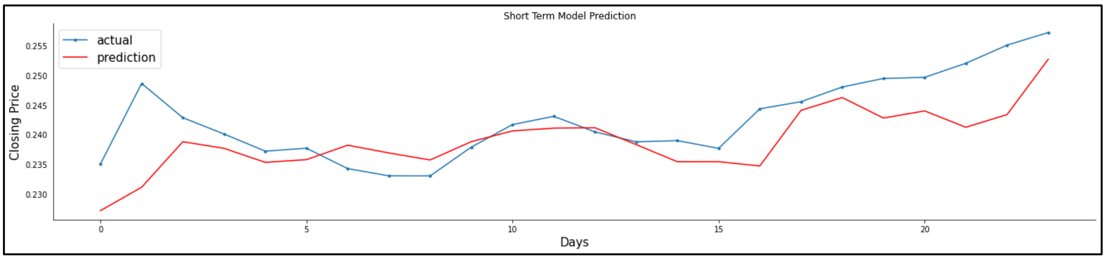


Figure 6.12: Model B Short Term – TASI Actual Close Price vs. Predicted Close.

6.2.1.3 Model C Evaluations

6.2.1.3.1 Optimizing the Window Size

All other parameters were kept a constant value and observed the behavior of the Mean Squared Error with the variation of Window Size to obtain a precise window size in days. The experiment is performed 10 times and Mean Squared Error is averaged out at the end. Figure 6.1 shows the variation of MSE with the variation of Window Size. It shows that a moving window size of 1 is giving the best accurate results. It hints that tomorrow's close price only depends mostly on yesterday's close price.

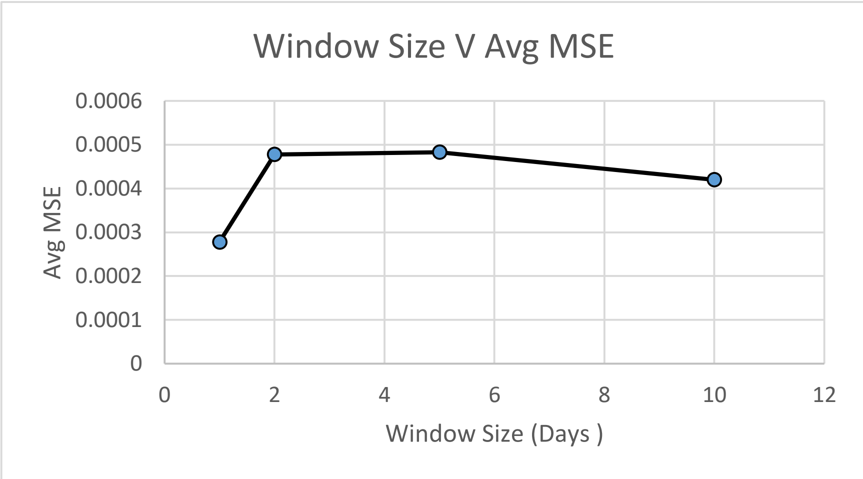


Figure 6.13: Model C –Short Term Window Size vs. MSE Variation

6.2.1.3.2 Optimizing the Neuron Count in Each Layer

Neuron Count in each layer was also tuned by keeping all other parameters constant and running the test 10 separate times to record the average performance. Figure 6.14 show that best accuracies are obtained when the neuron count per layer is between 20 and 30 for the model which feeds OHLC and Technical data via PCA Layer to the Neural Network (Model C).

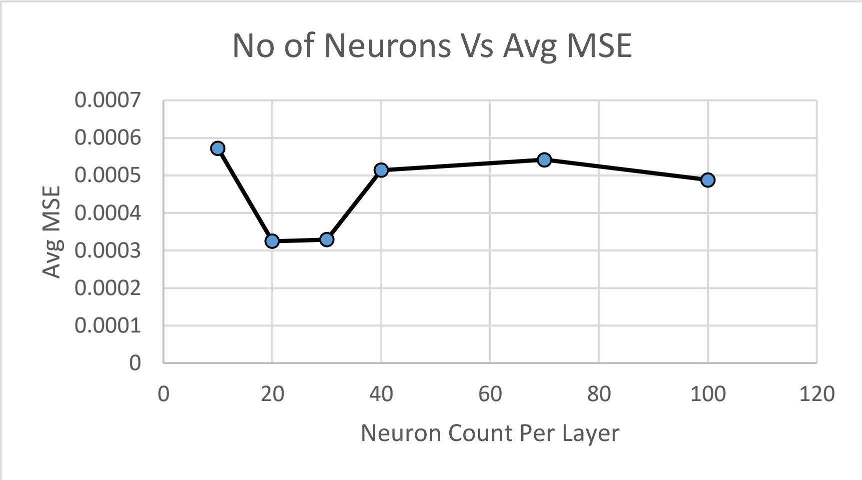


Figure 6.14: Model C –Short Term Neuron Count vs. MSE Variation

6.2.1.3.3 Optimizing the training Batch Size of the Network

Optimum Batch Size was found by keeping all other parameters in a constant value. Batch Size search was done following a grid search methodology. Each batch size point was evaluated repeatedly 10 times to obtain the averaged Mean Squared Error. Figure 6.15 show the batch size of 16 achieved the best results for the model which feeds OHLC and Technical Indicator data via a PCA Layer to the Neural Network (Model C).

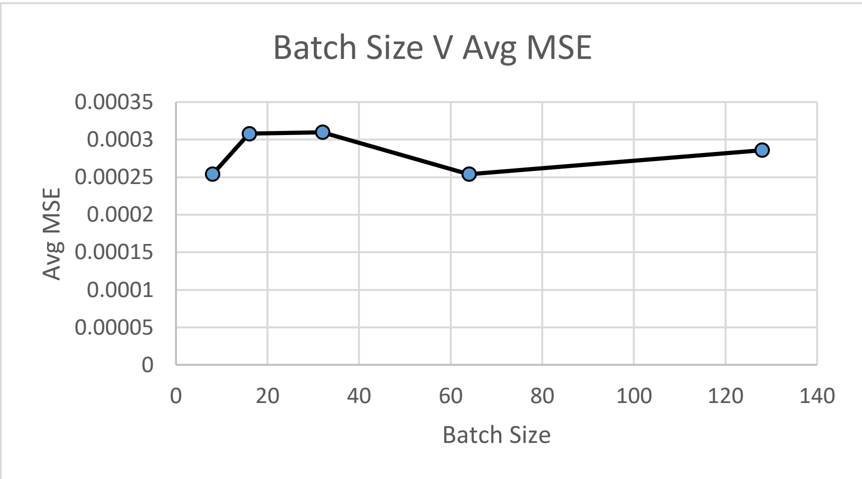


Figure 6.15: Model C –Short Term Batch Size vs. MSE Variation

6.2.1.3.4 Deep Neural Network Layer Count Tuning

Model which feeds OHLC data and Technical Indicator data via PCA Layer to the Neural Network (Model C) is trained and tested with All LSTM layers, All GRU layers and Hybrid of LSTM and GRU layers. Figure 6.15 and Figure 6.16 shows the variation of Mean Squared Error with No of LSTM and GRU Layer count. Best Performance for both LSTM and GRU networks were given when the layer count is between 1 and 5.

<u>LSTM Layer Count versus Mean Squared Error Variation</u>

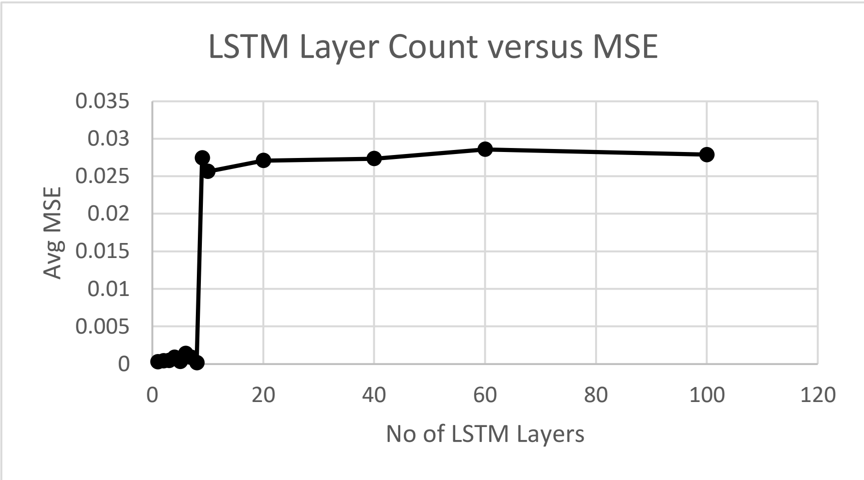


Figure 6.15: Model C –Short Term LSTM Layer Count vs. MSE Variation

<u>GRU Layer Count versus Mean Squared Error Variation</u>

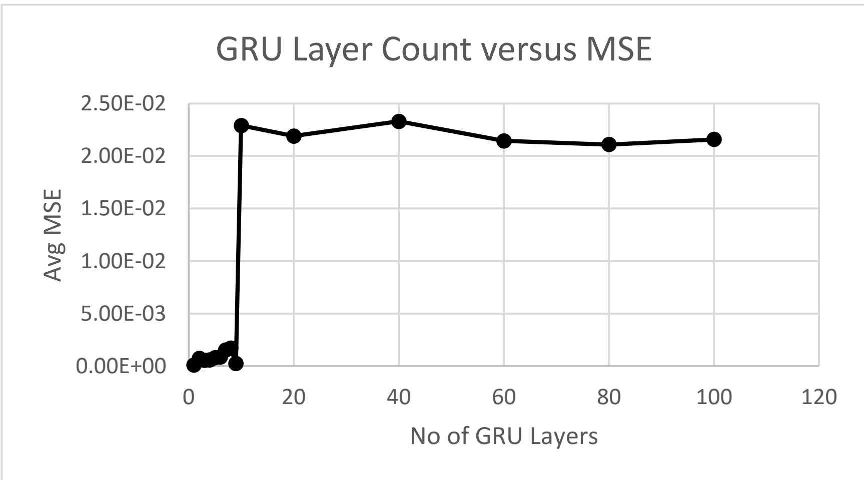


Figure 6.16: Model C –Short Term GRU Layer Count vs. MSE Variation

6.2.1.3.5 Overall Model Performance

Model which feeds OHLC and Technical Indicator data via PCA Layer to the Neural Network (Model C) is trained with all LSTM, All GRU and Hybrid network with LSTM and GRU Layers. Table 6.5 tabulates the best results of each network. Hybrid Network which comprised of LSTM and GRU Layers are giving best performance in terms of Maximum Absolute Percentage Error in the Model D. Best Accuracy level was under 1% which is considered as very reliable in data prediction domain.

Table 6.5 TASI Index – Short Term Results of Model C

Model	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	0.0000425661	0.0065242704	0.0050835697	1.0670511722
GRU	0.0000334553	0.0057840608	0.0046677053	0.9730952978
LSTM + GRU	**0.0000309564**	**0.0055638500**	**0.0039677359**	**0.8298736810**

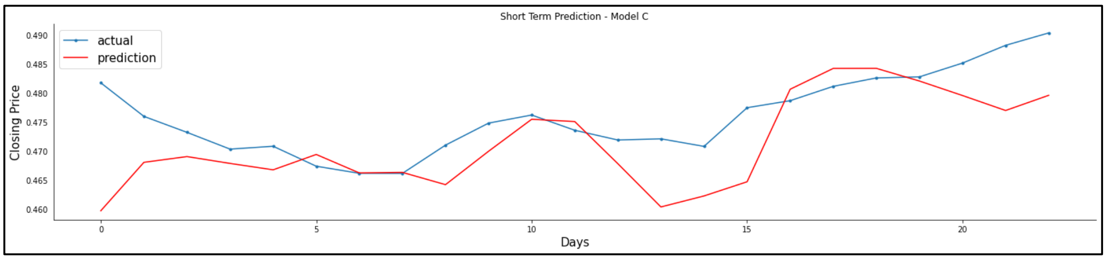


Figure 6.17: Model C Short Term – TASI Actual Close Price vs. Predicted Close

6.2.1.4 Model D Evaluations

6.2.1.4.1 Optimizing the Window Size

All other parameters were kept a constant value and observed the behavior of the Mean Squared Error with the variation of Window Size to obtain a precise window size in days. The experiment is performed 10 times and Mean Squared Error is averaged out at the end. Figure 6.18 shows the variation of MSE with the variation of Window Size. It shows that a moving window size of 10 is giving the best accurate results.

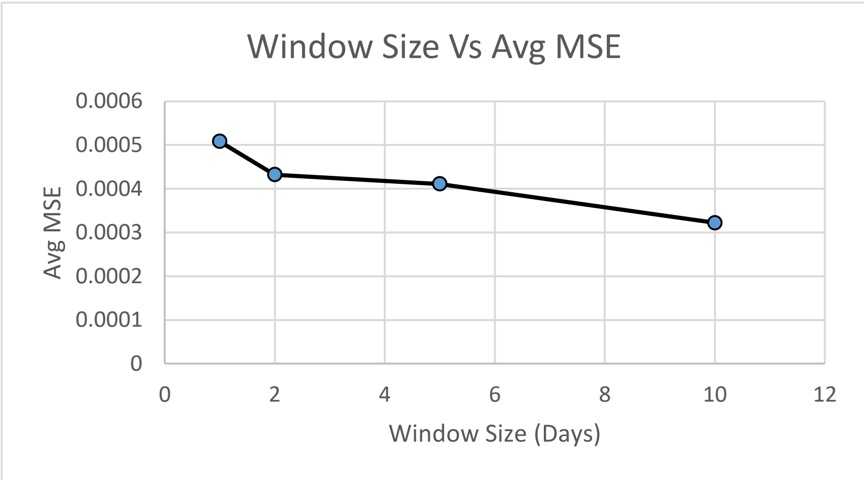


Figure 6.18: Model D –Short Term Window Size vs. MSE Variation

6.2.1.4.2 Optimizing the Neuron Count in Each Layer

Neuron Count in each layer was also tuned by keeping all other parameters constant and running the test 10 separate times to record the average performance. Figure 6.19 show that best accuracies are obtained when the neuron count per layer is between 10 and 30 for the model which feeds OHLC data directly and Technical Indicator data via a PCA Layer to the Neural Network (Model D).

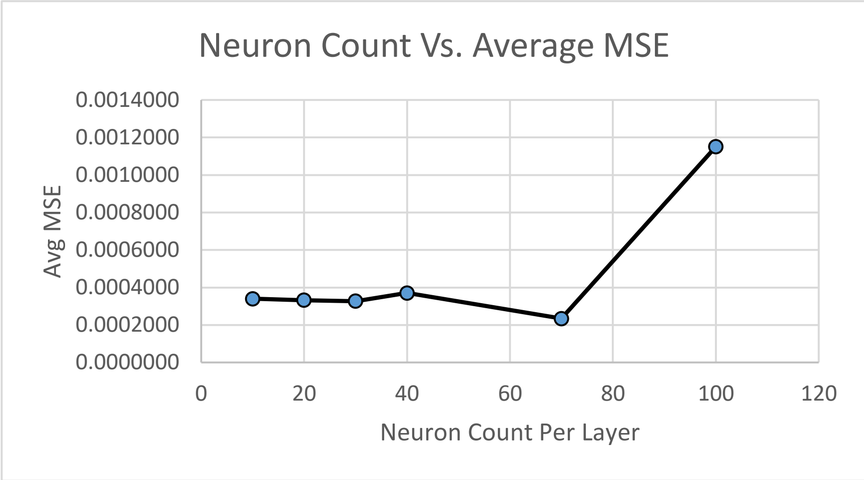


Figure 6.19: Model D –Short Term Neuron Count vs. MSE Variation

6.2.1.4.3 Optimizing the training Batch Size of the Network

Optimum Batch Size was found by keeping all other parameters in a constant value. Batch Size search was done following a grid search methodology. Each batch size point was evaluated repeatedly 10 times to obtain the averaged Mean Squared Error. Figure 6.20 show the batch size of 16 achieved the best results for the model which feeds OHLC data directly and Technical Indicator data via a PCA Layer to the Neural Network (Model D).

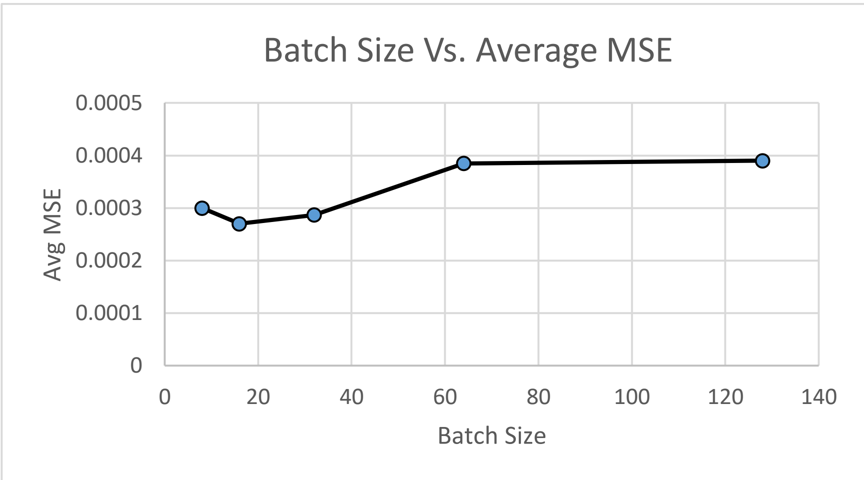


Figure 6.20: Model D –Short Term Batch Size vs. MSE Variation

6.2.1.4.4 Deep Neural Network Layer Count Tuning

Model which feeds OHLC data directly and Technical Indicator data via PCA Layer to the Neural Network (Model D) is trained and tested with All LSTM layers, All GRU layers and Hybrid of LSTM and GRU layers. Figure 6.21 and Figure 6.22 shows the variation of Mean Squared Error with No of LSTM and GRU Layer count. Best Performance for both LSTM and GRU networks were given when the layer count is between 1 and 4.

<u>LSTM Layer Count versus Mean Squared Error Variation</u>

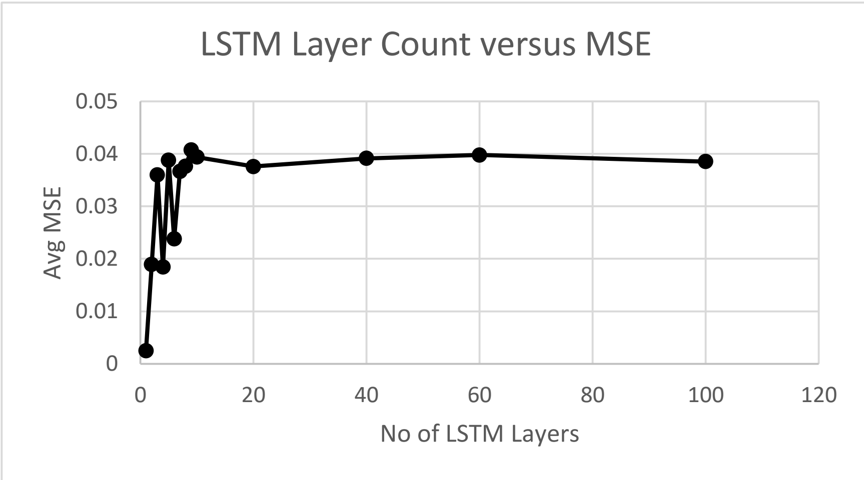


Figure 6.21: Model D –Short Term LSTM Layer Count vs. MSE Variation

<u>GRU Layer Count versus Mean Squared Error Variation</u>

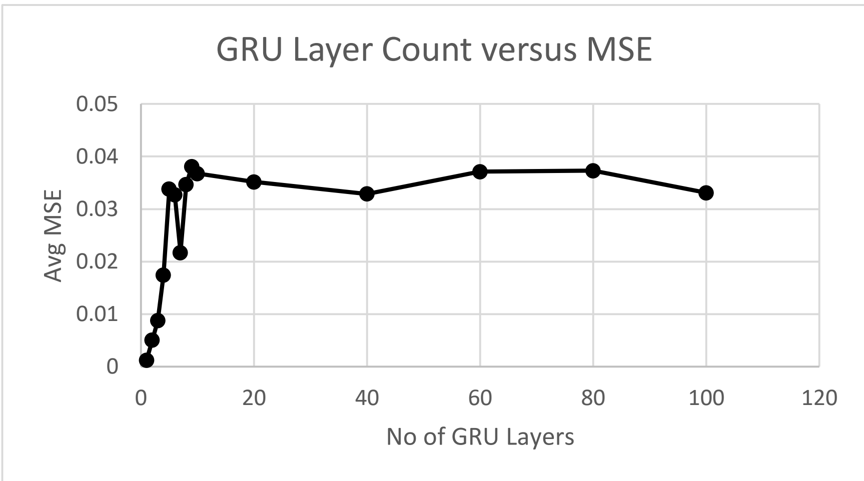


Figure 6.22: Model D –Short Term GRU Layer Count vs. MSE Variation

6.2.1.4.5 Overall Model Performance

Model which feeds OHLC data directly and Technical Indicator data via a PCA Layer to the Neural Network (Model D) is trained with all LSTM, All GRU and Hybrid network with LSTM and GRU Layers. Table 6.6 tabulates the best results of each network. Hybrid Network which comprised of LSTM and GRU Layers are giving best performance in terms of Maximum Absolute Percentage Error in the Model D. Best Accuracy level was under 1% which is considered as very reliable in data prediction domain.

Table 6.6 TASI Index – Short Term Results of Model D

Model	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	0.000052417770	0.007240011822	0.006201745942	1.303133010864
GRU	**0.000028631739**	**0.005350863095**	**0.004521274474**	**0.942254126072**
LSTM + GRU	0.000046300542	0.006804449949	0.005870713852	1.230605363846

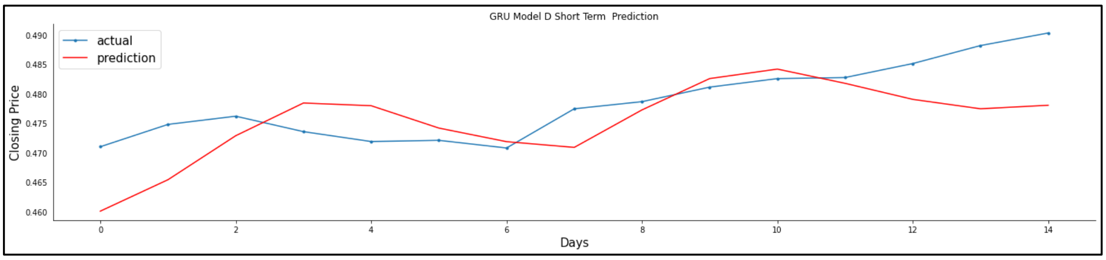


Figure 6.23: Model D Short Term – TASI Actual Close Price vs. Predicted Close

6.2.2 Stock Evaluations

All four models were trained with available data for each symbol and prediction was given on its closing price for next 25 days as Short term predictions. Results are stacked as below with each model. Charts have been attached with predicted vs. actual which gives a good visual comparison apart from quantitative analysis.

6.2.2.1 Model A Evaluations

Performance of the Five Symbols in Short Term are as follows. Short Term Prediction is used to predict the close price for the next 25 days into the future. As the Table 6.7 illustrates in bold letters, Symbol 2120, one out of five stocks has recorded its highest accuracy in short term with Model A. Mean Absolute Error Percentages are ranging from 1 % to 30% in the model which indicates that Model A is not very reliable one for stock predictions in short term.

Table 6.7: Model A – Short Term Stock Evaluations

Model	**Symbol Name**	**Mean Squared Error**	**Root Mean Squared Error**	**Mean Absolute Error**	**Mean Absolute Percentage Error**
LSTM	4050	0.00033236	0.01823085	0.01039002	2.66511917
GRU	4050	0.00035565	0.01885855	0.01117835	2.87517142
LSTM + GRU	4050	0.00018134	0.01346640	0.01082463	2.89739585
LSTM	4040	0.00006543	0.00808912	0.00557865	2.19815850
GRU	4040	0.00007016	0.00837610	0.00614884	2.33243966
LSTM + GRU	4040	0.00005632	0.00750448	0.00525159	1.63667357

LSTM	**2120**	**0.00001897**	**0.00435500**	**0.00328753**	**1.23123884**
GRU	2120	0.00008858	0.00941144	0.00835984	3.10739970
LSTM + GRU	2120	0.00002680	0.00517659	0.00400201	1.49995553
LSTM	2020	0.00464945	0.06818691	0.06571626	7.08480597
GRU	2020	0.00925344	0.09619482	0.09410292	10.14142513
LSTM + GRU	2020	0.06903801	0.26275086	0.26071736	28.15051270
LSTM	1150	0.00017543	0.01324482	0.01014208	1.11873043
GRU	1150	0.01374866	0.11725469	0.11377797	12.39693356
LSTM + GRU	1150	0.00969788	0.09847780	0.09649801	10.56657505

Figure 6.24 to Figure 6.28 visually illustrates the performance of Model A with each Stock Symbol. Figures have shown how Predicted Price from the Model A stacks up with Actual Price of the Stock.

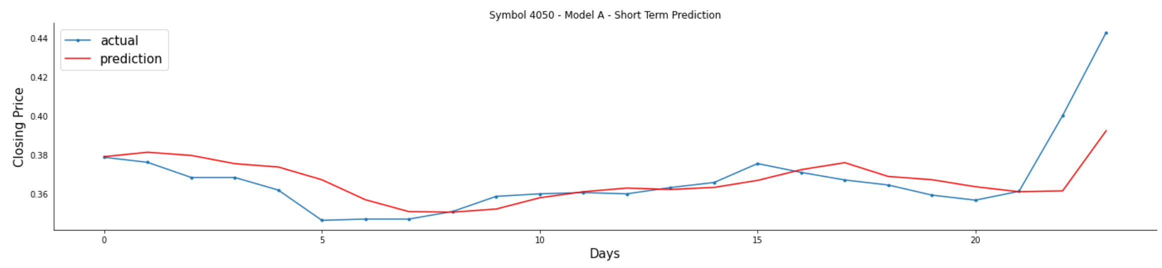


Figure 6.24: Model A - Symbol 4050 - Actual Close versus Predicted Close.

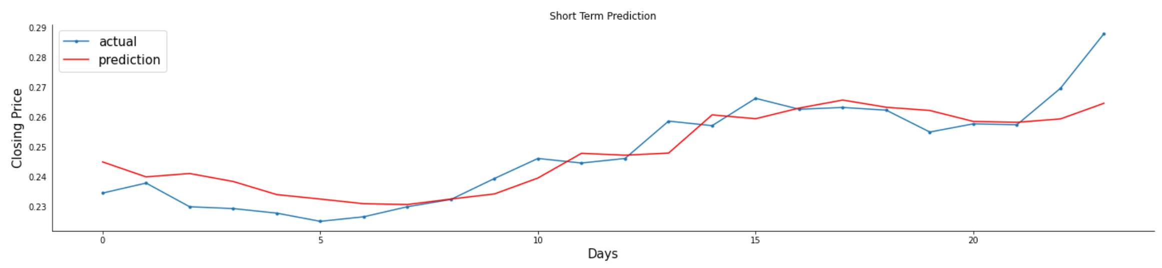


Figure 6.25: Model A - Symbol 4040 - Actual Close versus Predicted Close

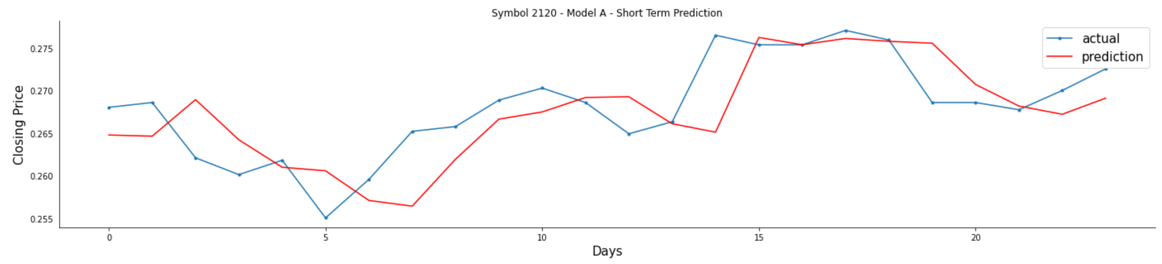


Figure 6.26: Model A - Symbol 2120 - Actual Close versus Predicted Close.

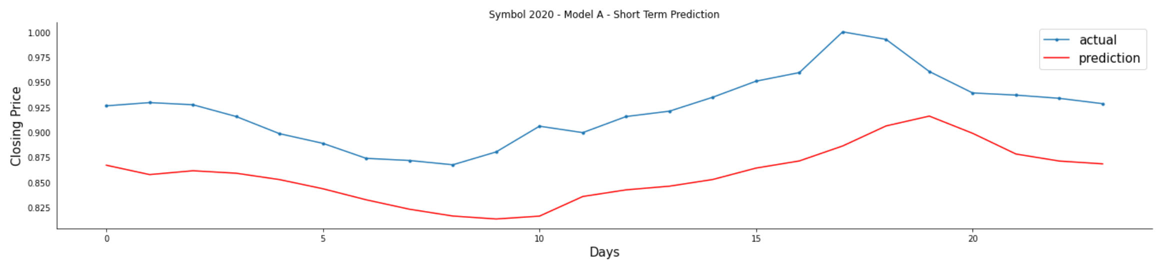


Figure 6.27: Model A - Symbol 2020 - Actual Close versus Predicted Close.

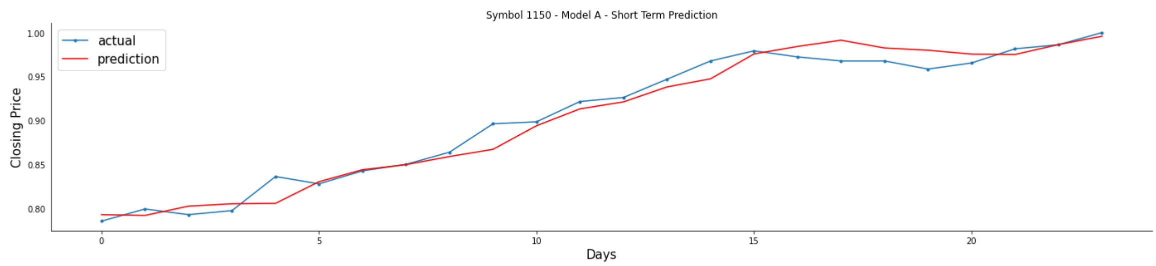


Figure 6.28: Model A - Symbol 1150 - Actual Close versus Predicted Close.

6.2.2.2 Model B Evaluations

Table 6.8 illustrates the Mean Absolute Error Percentage of the five stocks in short term predictions with Model B. Mean Absolute Error Percentages are ranging from 1 % to 30% in the model which indicates that Model B is not very reliable one for stock predictions in short term.

Table 6.8: Model B – Short Term Stock Evaluations

Model	Symbol Name	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	4050	0.00060115	0.02451828	0.02075505	5.67376328
GRU	4050	0.00075309	0.02744245	0.02303473	6.12843180
LSTM + GRU	4050	0.00020451	0.01430062	0.01064019	2.13981252
LSTM	4040	0.00082762	0.02876845	0.05672345	3.37472593
GRU	4040	0.00007625	0.00873191	0.00578349	2.00034729
LSTM + GRU	4040	0.00005623	0.00749881	0.00512453	1.52356292
LSTM	2120	0.00835001	0.09137838	0.09128759	34.10133362
GRU	2120	0.00414705	0.06439763	0.06426942	24.01856804
LSTM + GRU	2120	0.00059940	0.02448273	0.02354496	8.78253365

LSTM	2020	0.00185387	0.04305657	0.03773258	4.01835108
GRU	2020	0.00123713	0.03517279	0.03176783	3.48149967
LSTM + GRU	2020	0.00902081	0.09497792	0.09251349	10.11175346
LSTM	1150	0.00031922	0.01786660	0.01489062	1.65255260
GRU	1150	0.00088995	0.02983210	0.02392600	2.80241013
LSTM + GRU	1150	0.00150626	0.03881053	0.03472269	3.78921390

Figure 6.29 to Figure 6.33 visually illustrates the performance of Model B with each Stock Symbol. Figures have shown how Predicted Price from the Model B stacks up with Actual Price of the Stock. It also emphasizes the fact that most of the Model B predictions are not very accurate.

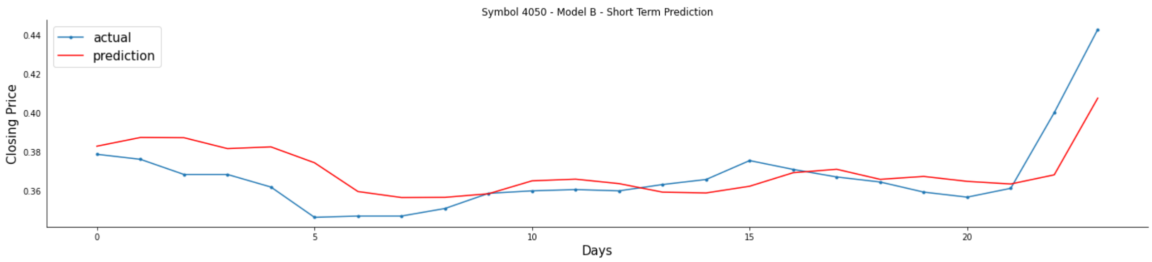


Figure 6.29: Model B - Symbol 4050 - Actual Close versus Predicted Close.

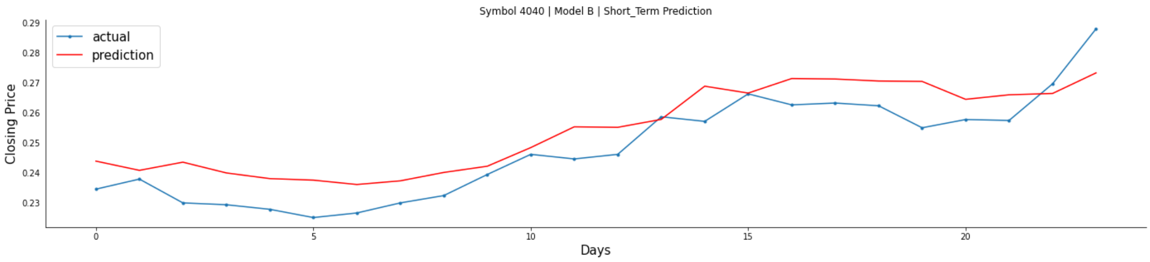


Figure 6.30: Model B - Symbol 4040 - Actual Close versus Predicted Close.

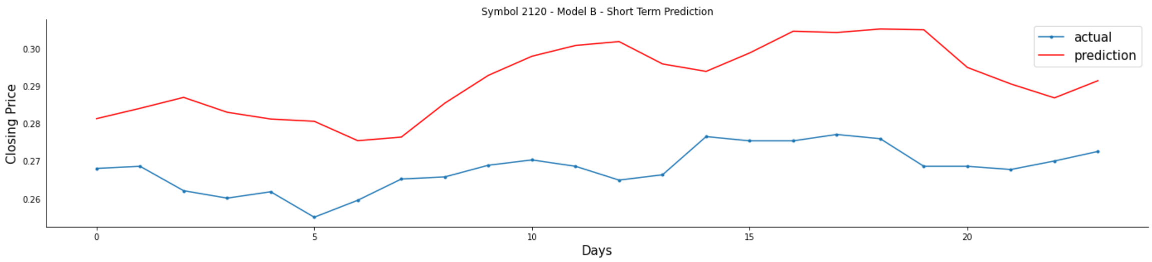


Figure 6.31: Model B - Symbol 2120 - Actual Close versus Predicted Close.

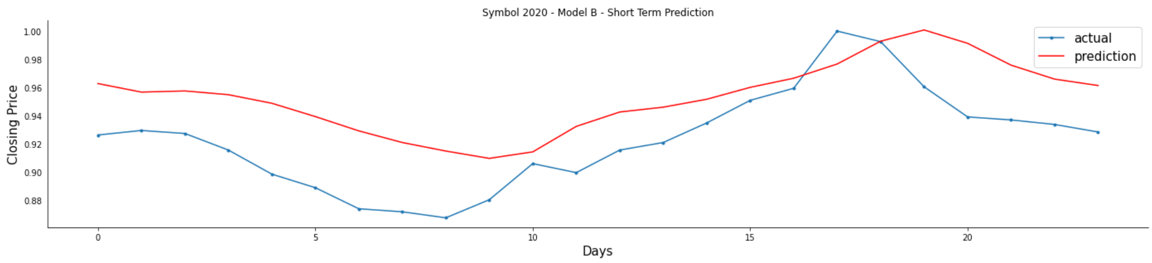


Figure 6.32: Model B - Symbol 2020 - Actual Close versus Predicted Close.

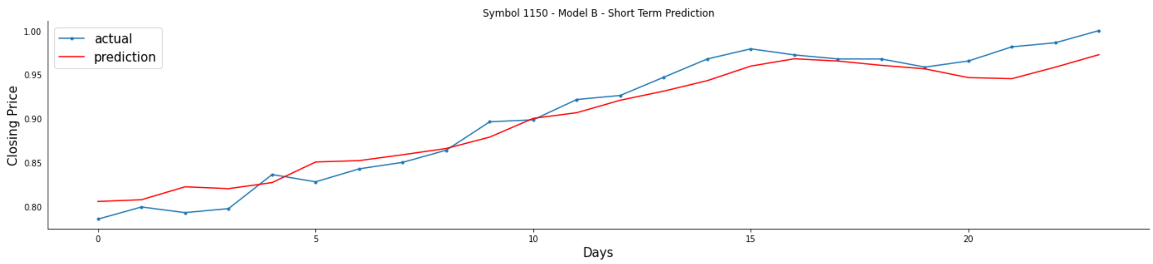


Figure 6.33: Model B - Symbol 1150 - Actual Close versus Predicted Close.

6.2.2.3 Model C Evaluations

Table 6.9 illustrates the Mean Absolute Error Percentage of the five stocks in short term predictions with model which feeds OHLC and Technical data via PCA Layer to the Neural Network (Model C). Mean Absolute Error Percentages are ranging from 1 % to 15% in the model which indicates that Model C is reliable most of the time apart from some instances. When Model C is applied with Hybrid Network the Error rates decline to 1%-5% range apart from stock 2020.

Table 6.9: Model C – Short Term Stock Evaluations

Model	Symbol Name	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	4050	0.00034229	0.01850109	0.01426512	3.80969119
GRU	4050	0.00026354	0.01623380	0.01274907	3.44115448
LSTM + GRU	4050	0.00030525	0.01747146	0.01058057	2.58756263
LSTM	4040	0.00006334	0.00795856	0.00632847	2.17634822
GRU	4040	0.00006378	0.00798645	0.00673235	1.98359247
LSTM + GRU	4040	0.00005221	0.00722578	0.00513477	1.50437292
LSTM	2120	0.00081502	0.02854861	0.02801439	10.45321751
GRU	2120	0.00106878	0.03269228	0.03228613	12.05662537
LSTM + GRU	2120	0.00015040	0.01226371	0.00977592	3.62864709

LSTM	2020	0.00183259	0.04280876	0.03966292	4.33129692
GRU	2020	0.00523520	0.07235472	0.06750068	7.22331190
LSTM + GRU	2020	0.02163161	0.14707689	0.14428294	15.53940296
LSTM	1150	0.00047692	0.02183845	0.01853216	1.99297965
GRU	1150	0.00206129	0.04540141	0.03998454	4.26033640
LSTM + GRU	1150	0.00283314	0.05322721	0.04679279	4.98128510

Figure 6.34 to Figure 6.38 visually illustrates the performance of Model C with each Stock Symbol. Figures have shown how Predicted Price from the Model C stacks up with Actual Price of the Stock.

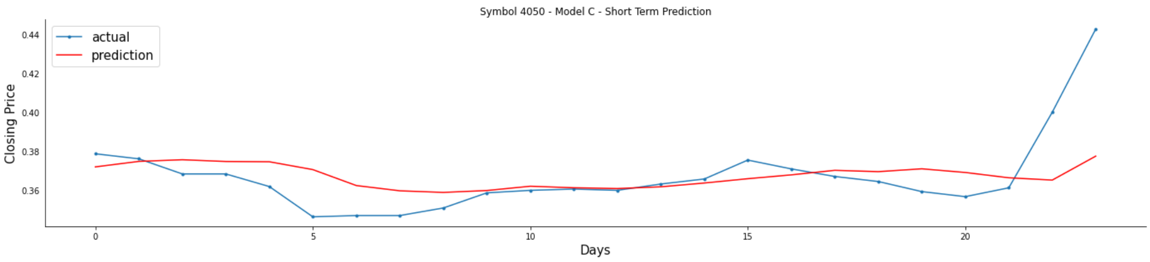


Figure 6.34: Model C - Symbol 4050 - Actual Close versus Predicted Close.

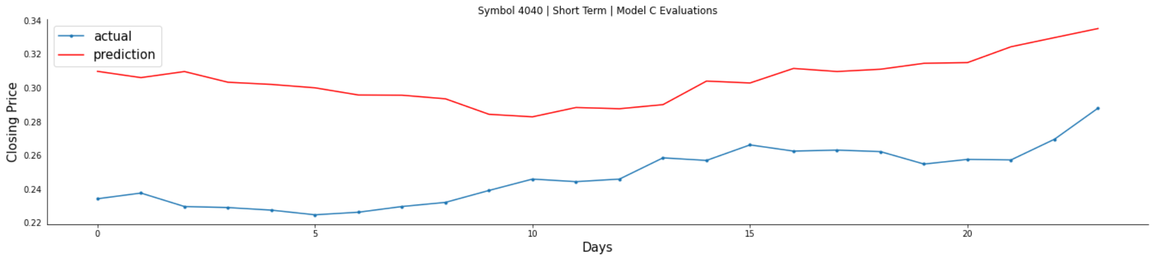


Figure 6.35: Model C - Symbol 4040 - Actual Close versus Predicted Close.

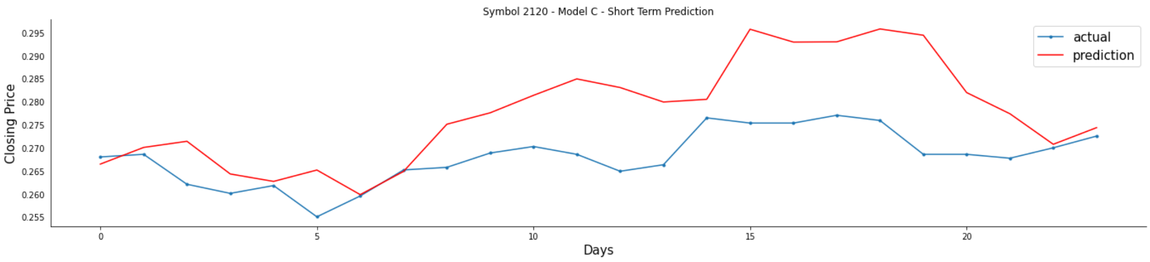


Figure 6.36: Model C - Symbol 2120 - Actual Close versus Predicted Close.

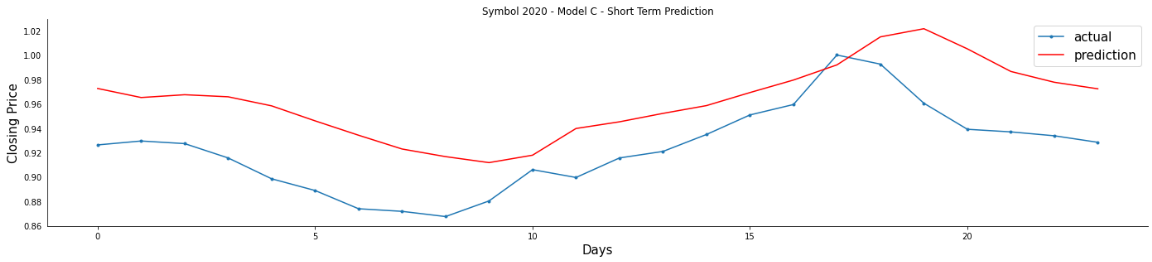


Figure 6.37: Model C - Symbol 2020 - Actual Close versus Predicted Close.

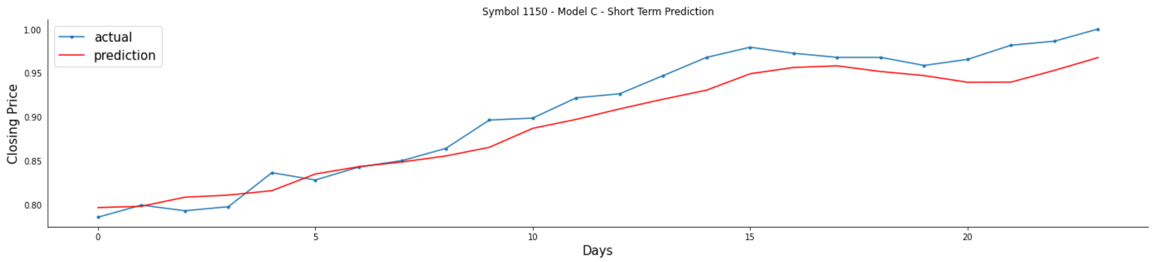


Figure 6.38: Model C - Symbol 1150 - Actual Close versus Predicted Close.

6.2.2.4 Model D Evaluations

As the Table 6.10 illustrates in bold letters, apart from Symbol 2120, four out of five stocks has recorded its highest accuracy in short term with Model D. Mean Absolute Error Percentages are ranging from 1 % to 5% in the model which indicates that Model D is very reliable for stock predictions in short term.

Table 6.10: Model D – Short Term Stock Evaluations

Model	**Symbol Name**	**Mean Squared Error**	**Root Mean Squared Error**	**Mean Absolute Error**	**Mean Absolute Percentage Error**
LSTM	4050	0.00020695	0.01438584	0.01053123	2.99862666
GRU	4050	0.00018267	0.01351539	0.01104159	2.97811580
LSTM + GRU	**4050**	**0.00022050**	**0.01484909**	**0.00891059**	**2.02160878**
LSTM	4040	0.00005638	0.00750886	0.00892345	2.27836499
GRU	4040	0.00005473	0.00739780	0.00902377	1.87353926
LSTM + GRU	**4040**	**0.00004689**	**0.00684780**	**0.00763648**	**1.23464945**
LSTM	2120	0.00018433	0.01357698	0.01093612	4.02582216
GRU	2120	0.00008937	0.00945374	0.00861915	3.18977666
LSTM + GRU	2120	0.00007638	0.00873963	0.00644048	2.38810968

LSTM	**2020**	**0.00043012**	**0.02073923**	**0.01480570**	**1.56872022**
GRU	2020	0.00493141	0.07022401	0.06797855	7.23180723
LSTM + GRU	2020	0.00053571	0.02314538	0.02110041	2.25423646
LSTM	**1150**	**0.000225618**	**0.015020572**	**0.010895002**	**1.135950923**
GRU	1150	0.001705059	0.041292358	0.036981974	4.253678799
LSTM + GRU	1150	0.001523649	0.039033942	0.032862227	3.483230591

Figure 6.39 to Figure 6.43 visually illustrates the performance of Model D with each Stock Symbol. Figures have shown how Predicted Price from the Model D stacks up with Actual Price of the Stock.

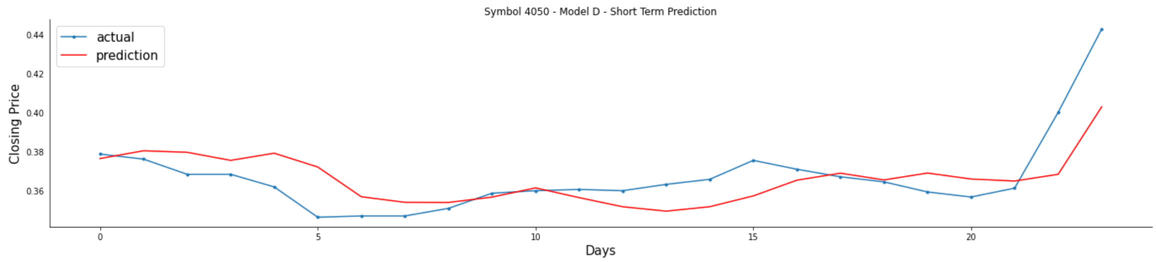


Figure 6.39: Model D - Symbol 4050 - Actual Close versus Predicted Close.

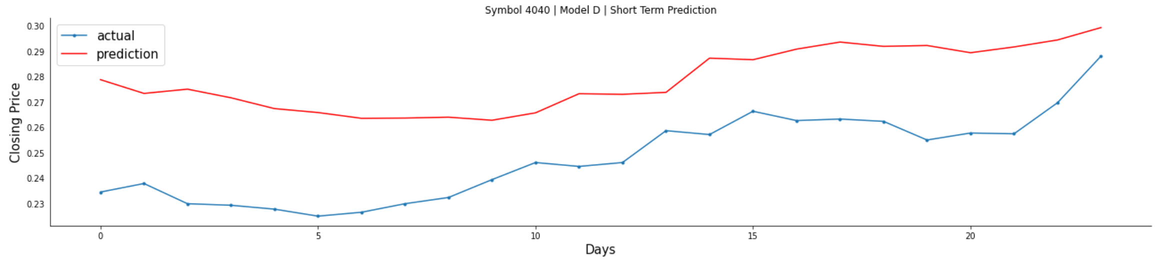


Figure 6.40: Model D - Symbol 4040 - Actual Close versus Predicted Close.

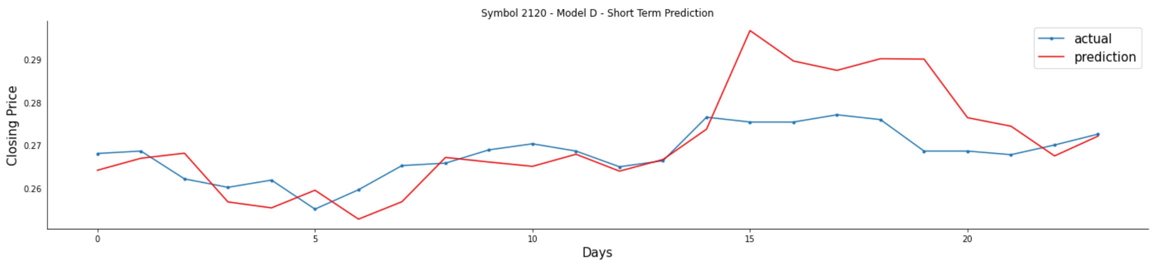


Figure 6.41: Model D - Symbol 2120 - Actual Close versus Predicted Close.

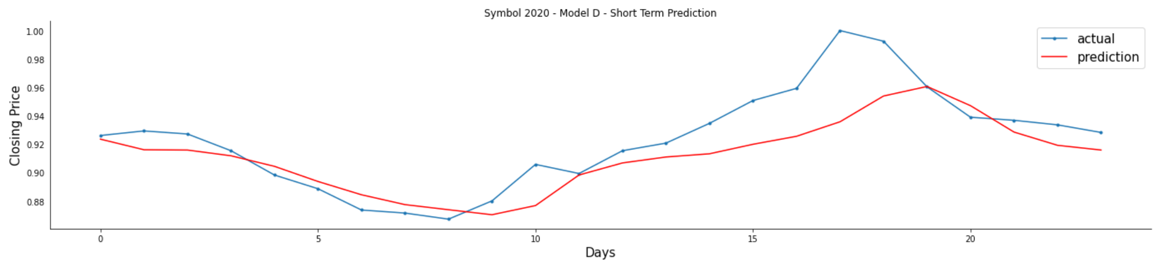


Figure 6.42: Model D - Symbol 2020 - Actual Close versus Predicted Close.

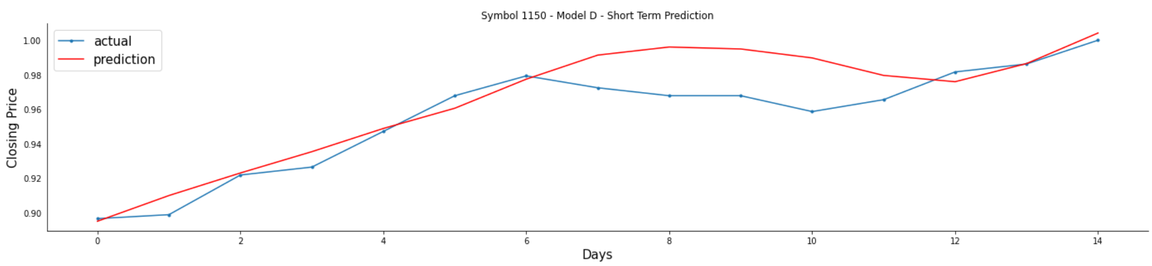


Figure 6.43: Model D - Symbol 1150 - Actual Close versus Predicted Close.

6.3 Medium Term Evaluation and Analysis

Medium Term evaluation is defined as predicting the close price behavior for next 100 day period. It is primarily important for retail customers and long term investors who anticipate returns in the medium term. Socio-economic factors do have an impact on the behavior of the prices but not as significant as in short term. Micro-economic quantitative analysis do explain a fair share of the price behavior in medium term.

6.3.1 Tadawul All Share Index (TASI) Evaluations

Prediction was done for Tadawul All Share Index commonly known as TASI. Results of the benchmark models for the Medium Term are as follows. Table 6.11 show that 3-Day Exponential Moving Average gave the lowest error rate percentage in Medium Term Predictions. Mean Absolute Percentage Error of 1.197% was set as the benchmark error percentage for the models presented in this research.

Table 6.11 TASI Index – Medium Term Results of the Bench Mark Models

Model	**MSE**	**RMSE**	**MAE**	**MAPE**
Moving Average	0.0000440293	0.0066354626	0.0054645642	1.25601713
Exponential Moving Average	**0.0000398222**	**0.0063104886**	**0.0052058081**	**1.19723460**
ARMA	0.0000848660	0.0439451832	0.0172583956	18.88023110
GARCH	1.2848395913	1.1335076494	0.9630897954	11.1322269785
SVM	0.0004801138	0.0219115012	0.0210159522	4.97003656
SVM	0.0022269816	0.0471909065	0.0470076419	10.94335887
FFNN	0.0001496288	0.0122322849	0.0108796209	5.11897611

6.3.1.1 Model A Evaluations

6.3.1.1.1 Optimizing the Window Size

All other parameters were kept a constant value and observed the behavior of the Mean Squared Error with the variation of Window Size to obtain a precise window size in days. Figure 6.44 shows that a window size of 1 achieved the lowest error rate. It meant that tomorrow close price is heavily dependent on today's close price in Model A where OHLC data is directly fed to the neural network.

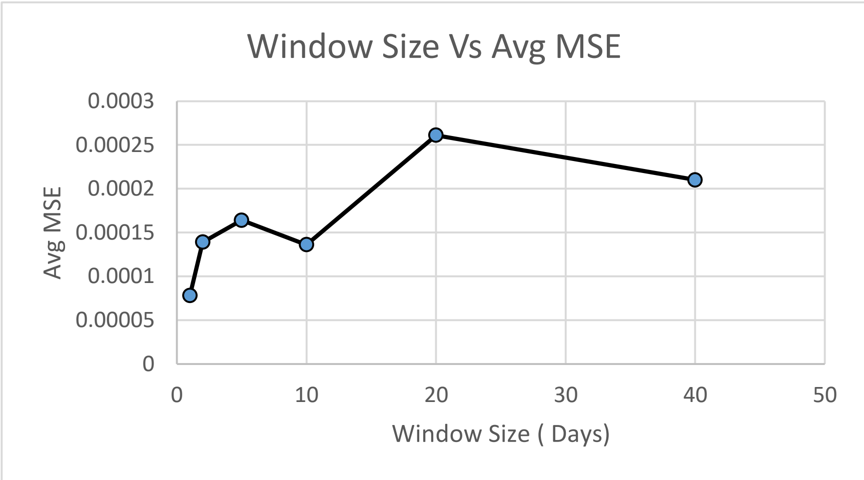


Figure 6.44: Model A –Medium Term Window Size Vs. MSE Variation

6.3.1.1.2 Optimizing the Neuron Count in Each Layer

Neuron Count in each layer was also tuned by keeping all other parameters constant and running the test 10 separate times to record the average Mean Square Error. Figure 6.45 show the variation between number of neurons per layer and MSE. It clearly indicates lower error rates when the neuron count ranges from 20 to 40 per layer.

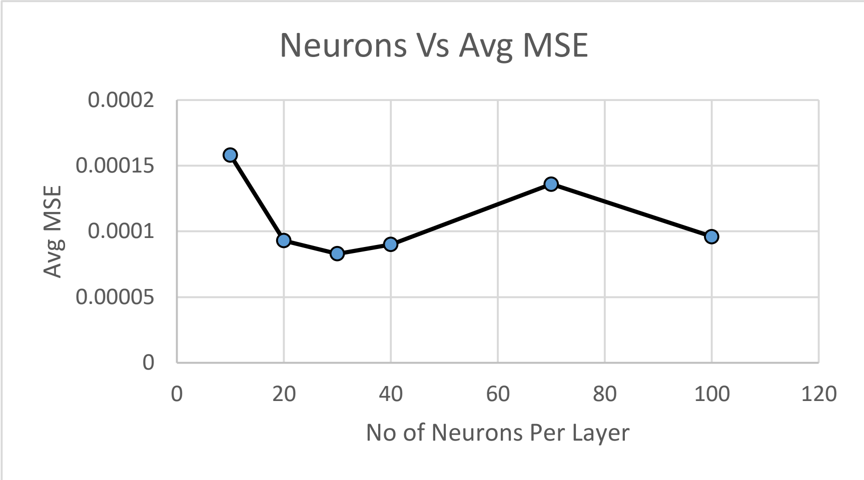


Figure 6.45: Model A –Medium Term Neuron Count Vs. MSE Variation

6.3.1.1.3 Optimizing the training Batch Size of the Network

Optimum Batch Size was found by keeping all other parameters in a constant value. Batch Size search was done following a grid search methodology. Each batch size point was evaluated repeatedly 10 times to obtain the averaged Mean Squared Error. Figure 6.46 shows the variation between Batch Size and MSE. It indicates better error rates for batch size of 16 and 64 for Model A where OHLC and Technical Indicator data are fed directly to the neural network.

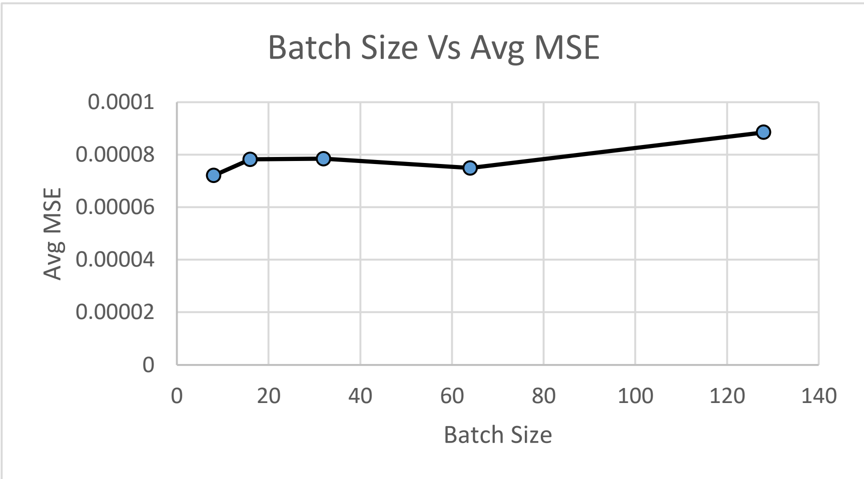


Figure 6.46: Model A –Medium Term Batch Size Vs. MSE Variation

6.3.1.1.4 Deep Neural Network Layer Count Tuning

Model which feeds OHLC data directly to the Neural Network (Model A) is trained and tested with All LSTM network and All GRU network. Figure 6.46 and Figure 6.47 shows the variation of Mean Squared Error with No of LSTM and GRU Layer count. Best Performance for both LSTM and GRU networks were given when the layer count is between 1 and 4.

<u>LSTM Layer Count versus Mean Squared Error Variation</u>

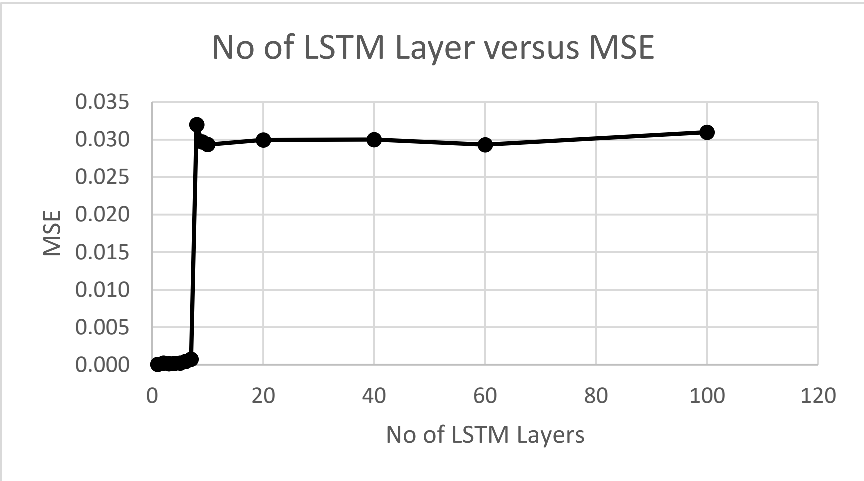


Figure 6.46: Model A –Medium Term LSTM Layer Count Vs. MSE Variation

<u>GRU Layer Count versus Mean Squared Error Variation</u>

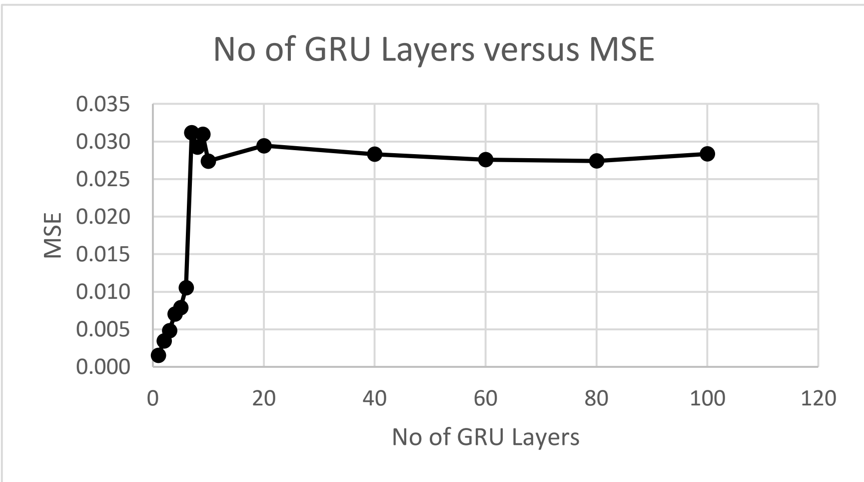


Figure 6.46: Model A –Medium Term GRU Layer Count Vs. MSE Variation

6.3.1.1.5 Overall Model Performance

Model which feeds OHLC data directly to the Neural Network (Model A) is trained with all LSTM, All GRU and Hybrid network with LSTM and GRU Layers. Table 6.12 tabulates the best results of each network. Hybrid Network which comprised of LSTM and GRU Layers was giving the best performance in terms of Maximum Absolute Percentage Error. MAPE of 1.52% is considered as reliable as it is closer to 1% benchmark. Figure 6.47 shows how Actual Close price varies with predicted close price over a period of 100 days. Model has given a good prediction over the period as per the plot between Actual and Predicted Close Prices.

Table 6.12 TASI Index – Medium Term Results of Model A

Model	**Mean Squared Error**	**Root Mean Squared Error**	**Mean Absolute Error**	**Mean Absolute Percentage Error**
LSTM	**0.00001818323**	**0.004264180083**	**0.003267641645**	**1.523129224777**
GRU	0.00004388102	0.006624275353	0.005406881683	2.550951957703
LSTM + GRU	0.00001882332	0.004338584840	0.003405641997	1.693543195724

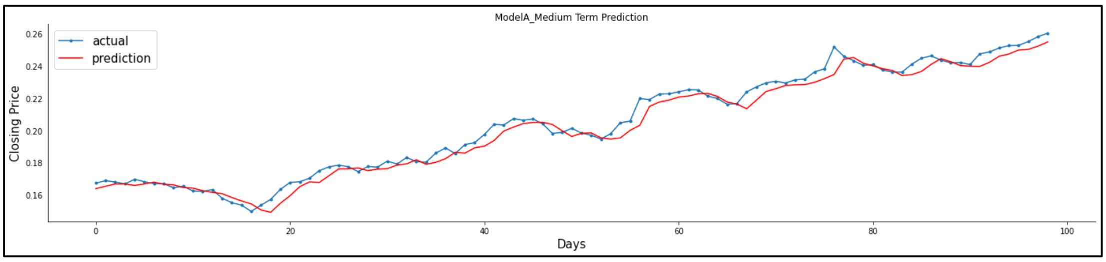


Figure 6.47: Model A Medium Term – TASI Actual Close vs. Predicted Close

6.3.1.2 Model B Evaluations

6.3.1.2.1 Optimizing the Window Size

All other parameters were kept a constant value and observed the behavior of the Mean Squared Error with the variation of Window Size to obtain a precise window size in days. The experiment is performed 10 times and Mean Squared Error is averaged out at the end. Figure 6.48 shows the variation of Window Size with MSE. Lowest error rates were indicated when Window Size is 5 days. It meant that tomorrow closing price tend to depend on closing prices of five previous days.

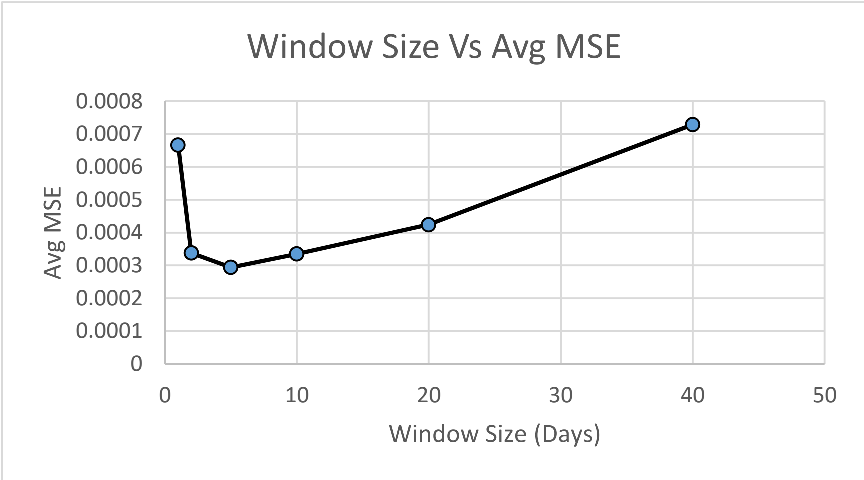


Figure 6.48: Model B –Medium Term Window Size Vs. MSE Variation

6.3.1.2.2 Optimizing the Neuron Count in Each Layer

Neuron Count in each layer was also tuned by keeping all other parameters constant and running the test 10 separate times to record the average Mean Square Error. Figure 6.49 show the variation between number of neurons per layer and MSE. It clearly indicates lower error rates when the neuron count is 20 neurons per layer.

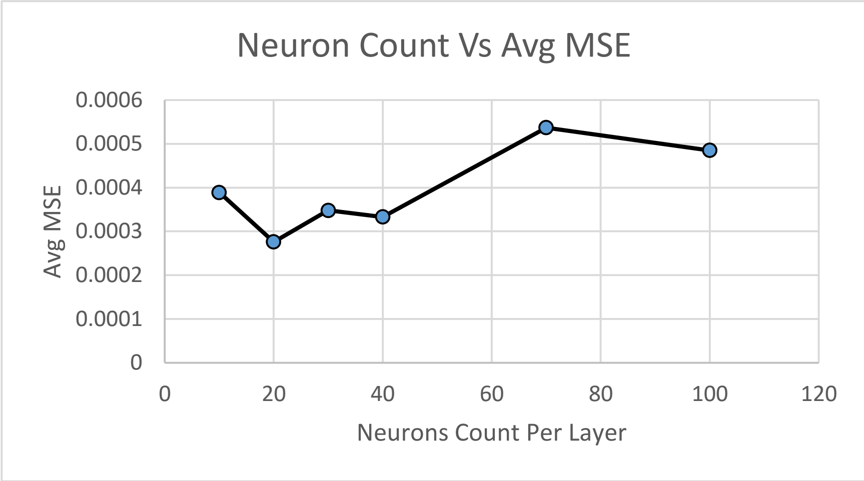


Figure 6.49: Model B –Medium Term Neuron Count vs. MSE Variation

6.3.1.2.3 Optimizing the training Batch Size of the Network

Optimum Batch Size was found by keeping all other parameters in a constant value. Batch Size search was done following a grid search methodology. Each batch size point was evaluated repeatedly 10 times to obtain the averaged Mean Squared Error. Figure 6.50 shows the variation between Batch Size and MSE. It indicates better error rates for batch size of 16 for Model B where OHLC and Technical Indicator data are fed directly to the neural network.

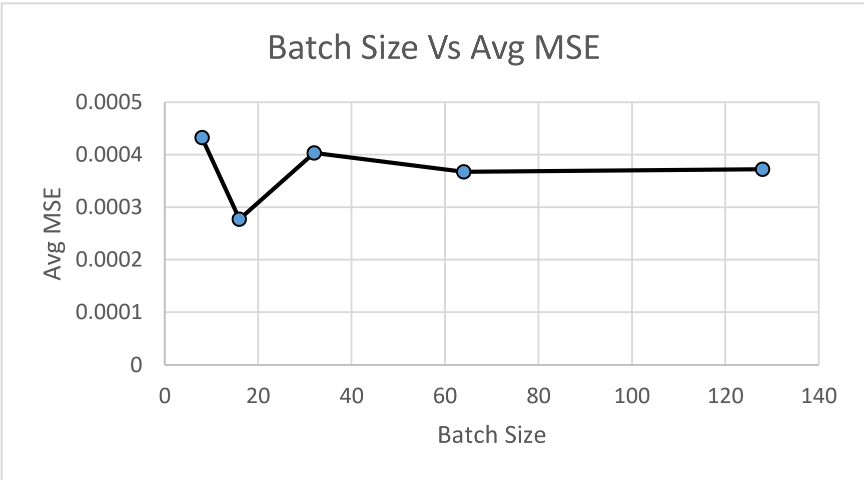


Figure 6.50: Model B –Medium Term Batch Size vs. MSE Variation

6.3.1.2.4 Deep Neural Network Layer Count Tuning

Model which feeds OHLC and Technical Indicator data directly to the Neural Network (Model B) is trained and tested with All LSTM network and All GRU network. Figure 6.51 and Figure 6.52 shows the variation of Mean Squared Error with No of LSTM and GRU Layer count. Best Performance for both LSTM and GRU networks were given when the layer count is between 1 and 5. Any number of layers above 5 showed very high error rates.

LSTM Layer Count versus Mean Squared Error Variation

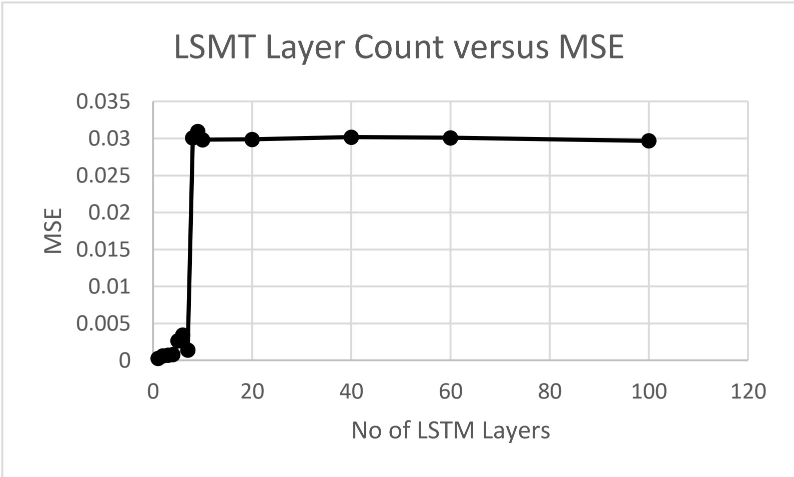


Figure 6.51: Model B –Medium Term LSTM Layer Count Vs. MSE Variation

GRU Layer Count versus Mean Squared Error Variation

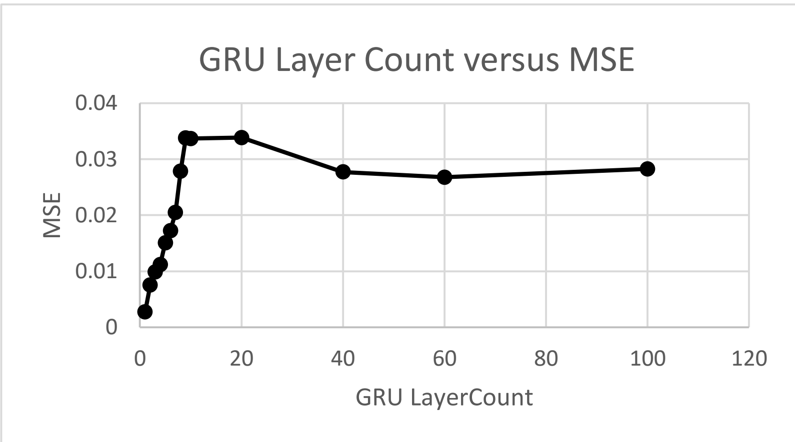


Figure 6.52: Model B –Medium Term GRU Layer Count vs. MSE Variation

6.3.1.2.5 Overall Model Performance

Model which feeds OHLC and Technical Indicator data directly to the Neural Network (Model B) is trained with all LSTM, All GRU and Hybrid network with LSTM and GRU Layers. Table 6.13 tabulates the best results of each network. Hybrid Network which comprised of LSTM and GRU Layers was giving the best performance in terms of Maximum Absolute Percentage Error. MAPE of 2.10% is considered as reliable as it is closer to 1% benchmark. But it is less accurate than the MAPE of Model A which was previously evaluated. Figure 6.53 shows how Actual Close price varies with predicted close price from Model B over a period of 100 days.

Table 6.13 TASI Index – Medium Term Results of Model B

Model	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	0.0000298983	0.0054679299	0.0041690292	2.1175842285
GRU	**0.0000281287**	**0.005303652957**	**0.004318528809**	**2.105489253998**
LSTM + GRU	0.0000383245	0.0061906809	0.0047762268	2.4696993828

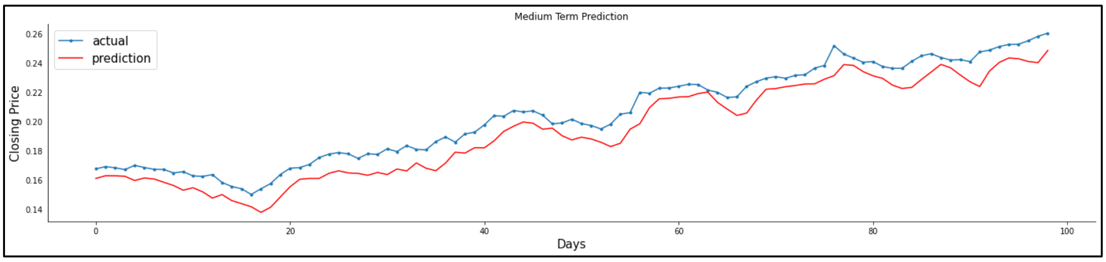


Figure 6.53: Model B Medium Term - TASI Actual Close vs. Predicted Close.

6.3.1.3 Model C Evaluations

6.3.1.3.1 Optimizing the Window Size

All other parameters were kept a constant value and observed the behavior of the Mean Squared Error with the variation of Window Size to obtain a precise window size in days. The experiment is performed 10 times and Mean Squared Error is averaged out at the end. Figure 6.54 shows the variation of Window Size with MSE. Lowest error rates were indicated when Window Size is 1 day. It meant that tomorrow closing price tend to depend on closing price of previous day only.

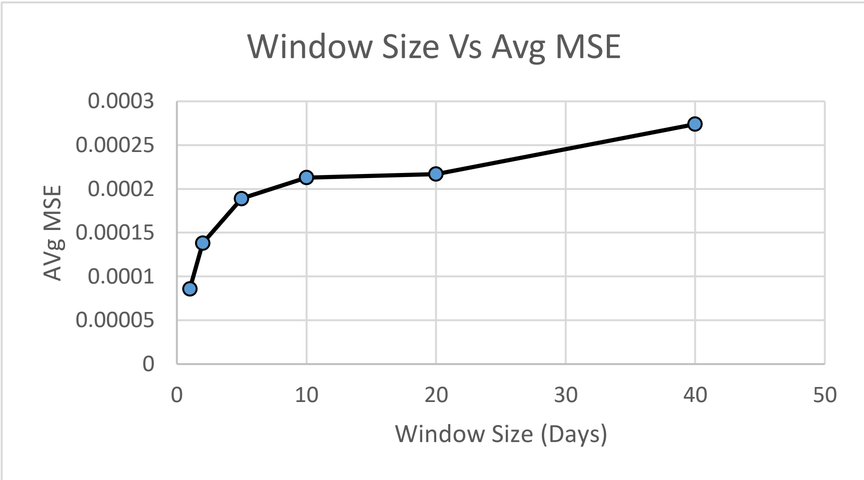


Figure 6.54: Model C –Medium Term Window Size vs. MSE Variation

6.3.1.3.2 Optimizing the Neuron Count in Each Layer

Neuron Count in each layer was also tuned by keeping all other parameters constant and running the test 10 separate times to record the average performance. Figure 6.55 show the variation between number of neurons per layer and MSE. It clearly indicates lower error rates when the neuron count is 10 and 20 neurons per layer.

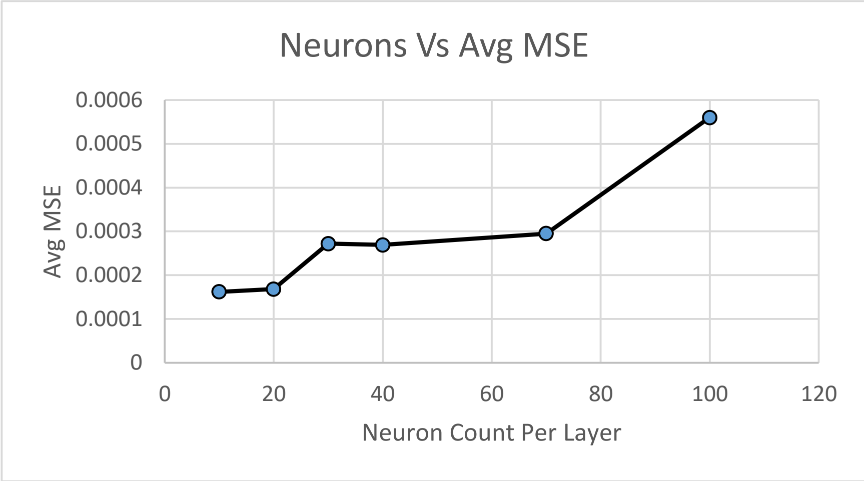


Figure 6.55: Model C –Medium Term Neuron Count vs. MSE Variation

6.3.1.2.3 Optimizing the training Batch Size of the Network

Optimum Batch Size was found by keeping all other parameters in a constant value. Batch Size search was done following a grid search methodology. Each batch size point was evaluated repeatedly 10 times to obtain the averaged Mean Squared Error. Figure 6.56 shows the variation between Batch Size and MSE. It indicates better error rates for batch size of 16 for Model C where OHLC and Technical Indicator data are fed via PCA Layer to the neural network.

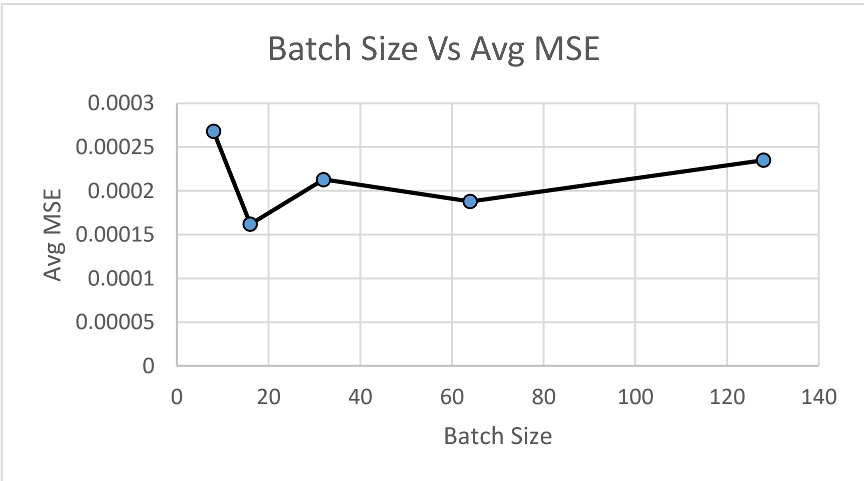


Figure 6.56: Model C –Medium Term Batch Size vs. MSE Variation

6.3.1.3.4 Deep Neural Network Layer Count Tuning

Model which feeds OHLC and Technical Indicator data via PCA Layer to the Neural Network (Model C) is trained and tested with All LSTM network and All GRU network. Figure 6.57 and Figure 6.58 shows the variation of Mean Squared Error with No of LSTM and GRU Layer count. Best Performance in GRU networks were given when the layer count is between 1 and 3 and 1 to 4 neurons per layer in LSTM Layers. Any number of layers above 5 showed very high error rates.

LSTM Layer Count versus Mean Squared Error Variation

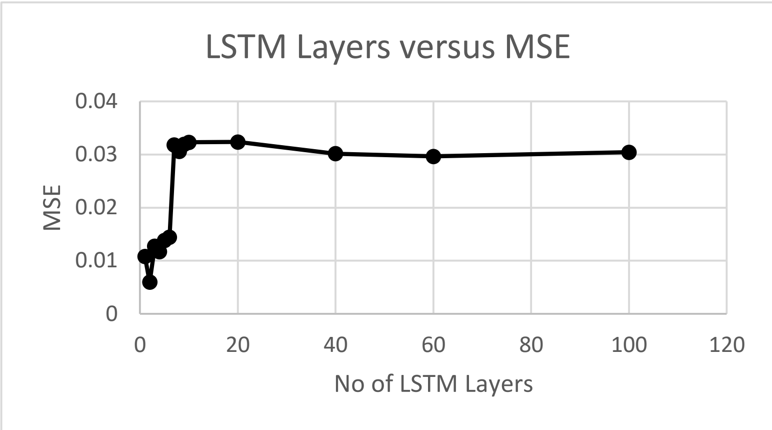


Figure 6.57: Model C –Medium Term LSTM Layer Count vs. MSE Variation

GRU Layer Count versus Mean Squared Error Variation

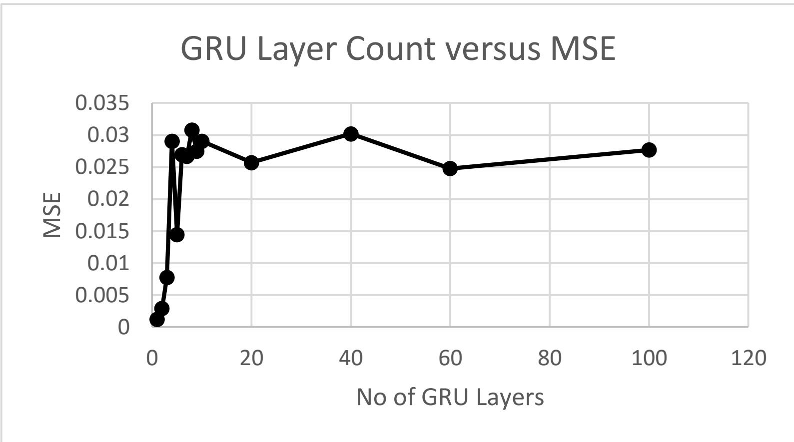


Figure 6.58: Model C –Medium Term GRU Layer Count vs. MSE Variation

6.3.1.3.5 Overall Model Performance

Model which feeds OHLC and Technical Indicator data via PCA Layer to the Neural Network (Model C) is trained with all LSTM, All GRU and a Hybrid network with LSTM and GRU Layers. Table 6.14 tabulates the best results of each network. Hybrid Network which comprised of LSTM and GRU Layers was giving the best performance in terms of Maximum Absolute Percentage Error. MAPE of 0.84% is considered as very reliable as it is below to 1% academic benchmark. Figure 6.59 shows how Actual Close price varies with predicted close price calculated from Model C over a period of 100 days.

Table 6.14 TASI Index – Medium Term Results of Model C

Model	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	0.0000332111	0.0057629151	0.0046549551	1.0689924955
GRU	0.0000333540	0.0057752942	0.0046518286	1.0556257963
LSTM + GRU	**0.0000233648**	**0.0048337234**	**0.0037286903**	**0.8486987352**

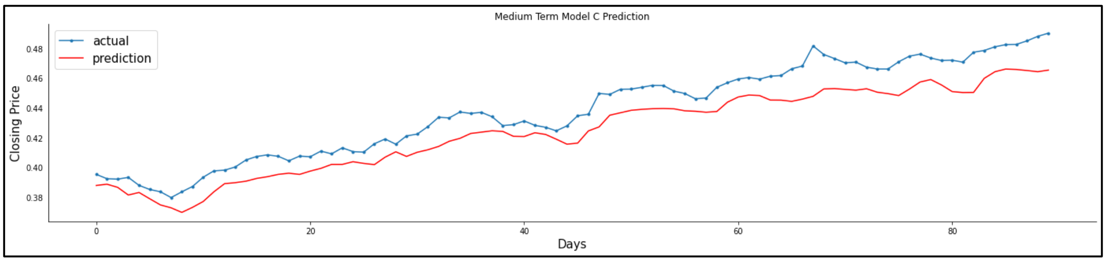


Figure 6.59: Model C Medium Term – TASI Actual Close vs. Predicted Close.

6.3.1.4 Model D Evaluations

6.3.1.4.1 Optimizing the Window Size

All other parameters were kept a constant value and observed the behavior of the Mean Squared Error with the variation of Window Size to obtain a precise window size in days. The experiment is performed 10 times and Mean Squared Error is averaged out at the end. Figure 6.60 shows the variation of Window Size with MSE. Lowest error rates were indicated when Window Size is 5 and 10 days.

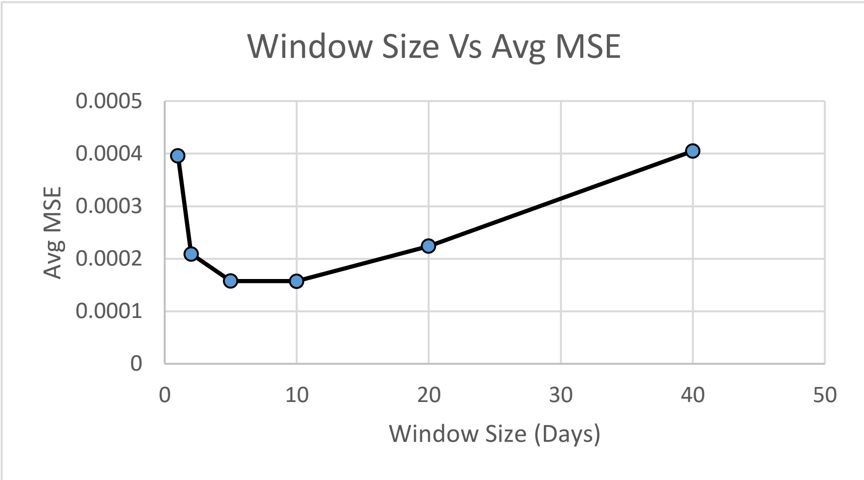


Figure 6.60: Model D –Medium Term Window Size vs. MSE Variation

6.3.1.4.2 Optimizing the Neuron Count in Each Layer

Neuron Count in each layer was also tuned by keeping all other parameters constant and running the test 10 separate times to record the average performance. Figure 6.61 show the variation between number of neurons per layer and MSE. It clearly indicates lower error rates when the neuron count is in the range of 10 to 30 neurons per layer.

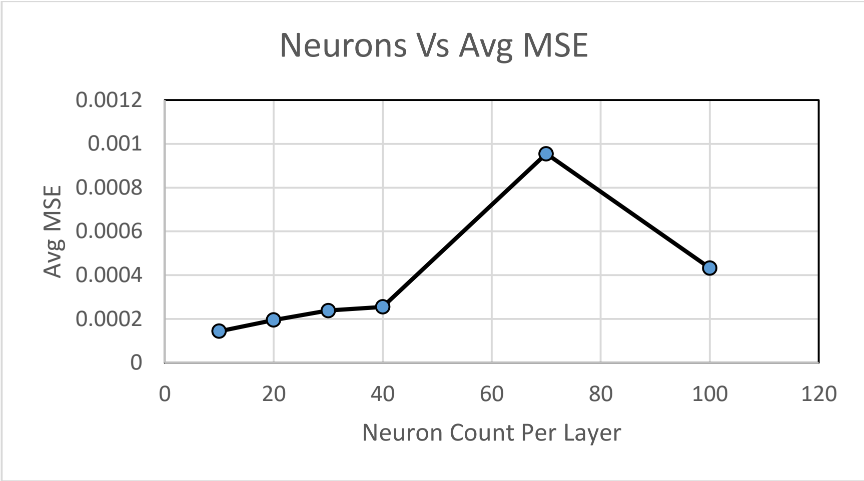


Figure 6.61: Model D –Medium Term Neuron Count vs. MSE Variation

6.3.1.4.3 Optimizing the training Batch Size of the Network

Optimum Batch Size was found by keeping all other parameters in a constant value. Batch Size search was done following a grid search methodology. Each batch size point was evaluated repeatedly 10 times to obtain the averaged Mean Squared Error. Figure 6.56 shows the variation between Batch Size and MSE. It indicates better error rates for batch size of 64 for Model D where OHLC data is fed directly and Technical Indicator data is fed via PCA Layer to the neural network.

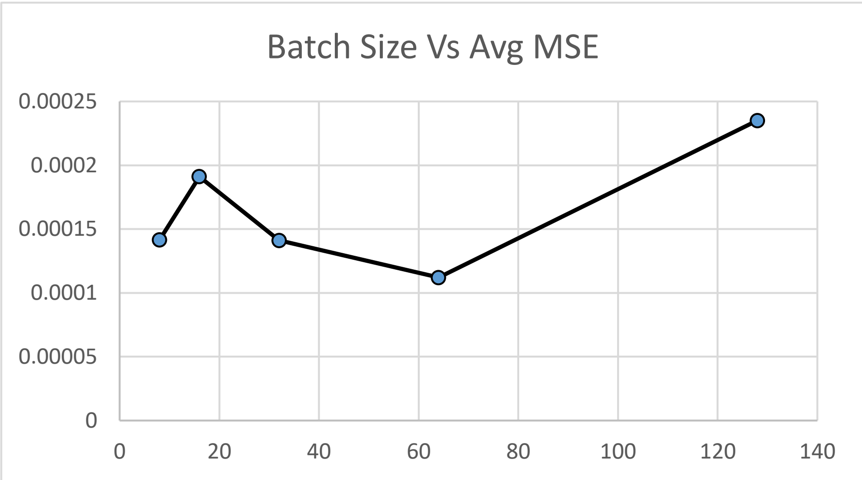


Figure 6.62: Model D –Medium Term Batch Size vs. MSE Variation

6.3.1.4.4 Deep Neural Network Layer Count Tuning

Model which feeds OHLC data directly and Technical Indicator data via PCA Layer to the Neural Network (Model D) is trained and tested with All LSTM network and All GRU network. Figure 6.63 and Figure 6.64 shows the variation of Mean Squared Error with No of LSTM and GRU Layer count. Best Performance in both LSTM and GRU networks were given when the layer count is between 1 and 4. Any number of layers above 4 showed very high error rates.

LSTM Layer Count versus Mean Squared Error Variation

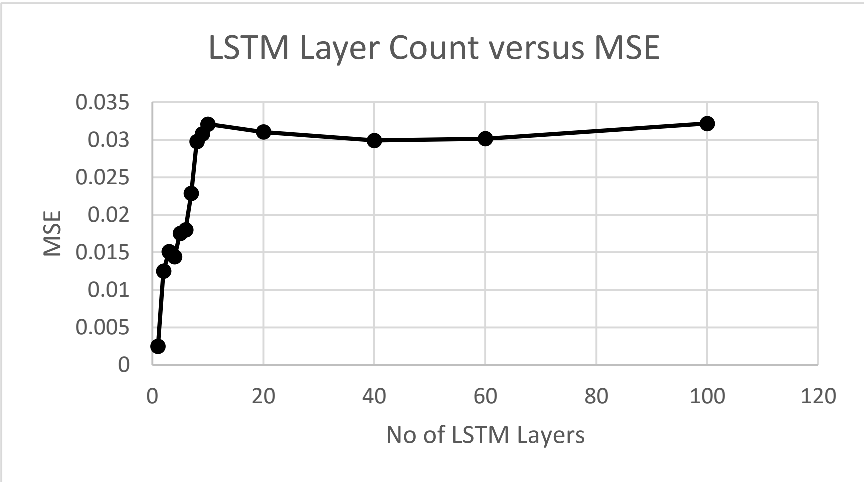


Figure 6.63: Model D –Medium Term LSTM Layer Count vs. MSE Variation

GRU Layer Count versus Mean Squared Error Variation

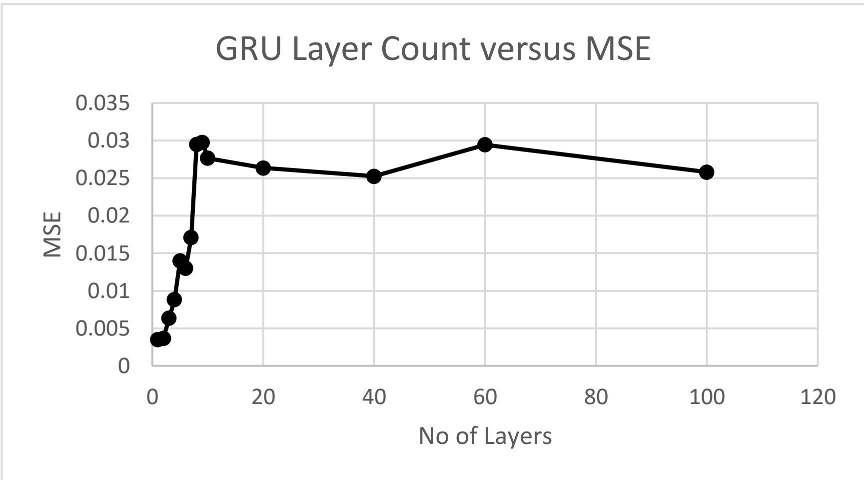


Figure 6.64: Model D –Medium Term GRU Layer Count vs. MSE Variation

6.3.1.4.5 Overall Model Performance

Model which feeds OHLC data directly and Technical Indicator data via PCA Layer to the Neural Network (Model D) is trained with all LSTM, All GRU and a Hybrid network with LSTM and GRU Layers. Table 6.15 tabulates the best results of each network. Hybrid Network which comprised of LSTM and GRU Layers was giving the best performance in terms of Maximum Absolute Percentage Error. MAPE of 1.15% is considered as very reliable as it is close to 1% academic benchmark. Figure 6.65 shows how Actual Close price varies with predicted close price calculated from Model D over a period of 100 days.

Table 6.15 TASI Index – Medium Term Results of Model D

Model	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	0.00004582372	0.00676932185	0.00549339503	1.26796782016
GRU	0.00003988464	0.00631542876	0.005163199268	1.17833602428
LSTM + GRU	**0.00004441334**	**0.00666433339**	**0.00522491848**	**1.15255343914**

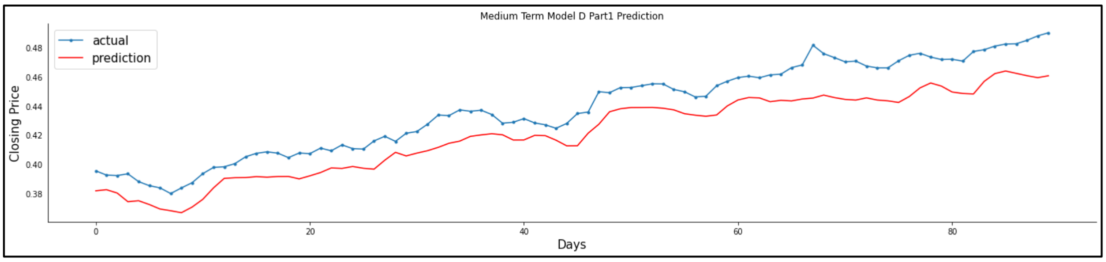


Figure 6.65: Model D Medium Term - TASI Actual Close vs. Predicted Close.

6.3.2 Stock Evaluations

All four models were trained with available data for each symbol and prediction was given on its closing price for next 100 days as Medium term predictions. Results are stacked as below with each model. Charts have been attached with predicted vs. actual which gives a good visual comparison apart from quantitative analysis.

6.3.2.1 Model A Evaluations

Performance of the Five Symbols in Medium Term are as follows. Medium Term Prediction is used to predict the close price for the next 100 days into the future. As the Table 6.16 illustrates in bold letters, Symbol 2120, one out of five stocks has recorded its lowest error rate in medium term with Model A. Best error rate of 2.19% for Symbol 2120 is a slight increase from its short term error rates. It should be anticipated as the prediction period now is 100 days not 25 days and the error rates are also expected to shoot up.

Table 6.16: Model A – Medium Term Stock Evaluations

Model	Symbol Name	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	4050	0.00018773	0.01311378	0.00813811	2.09347200
GRU	4050	0.00121624	0.02526556	0.02296364	5.74708748
LSTM + GRU	4050	0.00051180	0.01382626	0.01024821	2.50382376
LSTM	4040	0.00007638	0.08993954	0.00756376	2.92827204
GRU	4040	0.00008253	0.09152103	0.00743638	3.10273937

LSTM + GRU	4040	0.00006664	0.08662840	0.00677220	2.63393629
LSTM	**2120**	**0.00005813**	**0.00807649**	**0.00591873**	**2.19515872**
GRU	2120	0.00064772	0.02544308	0.02465708	9.01910019
LSTM + GRU	2120	0.00021824	0.01482092	0.01124226	4.09494734
LSTM	2020	0.00110597	0.03324319	0.02745982	3.03836775
GRU	2020	0.00075382	0.02744610	0.02113821	2.85683894
LSTM + GRU	2020	0.00239268	0.04916215	0.04244069	4.78009224
LSTM	1150	0.00911014	0.09421674	0.09149723	10.84615231
GRU	1150	0.01228164	0.09566188	0.09215062	10.91969395
LSTM + GRU	1150	0.01018023	0.08844626	0.08549073	10.14897060

Figure 6.66 to Figure 6.70 visually illustrates the performance of Model A with each Stock Symbol. Figures have shown how Predicted Price from the Model A stacks up with Actual Price of the Stock. The plot of Symbol 2120 which is achieved the highest medium term performance with Model A shows much closer predicted curve to the actual curve.

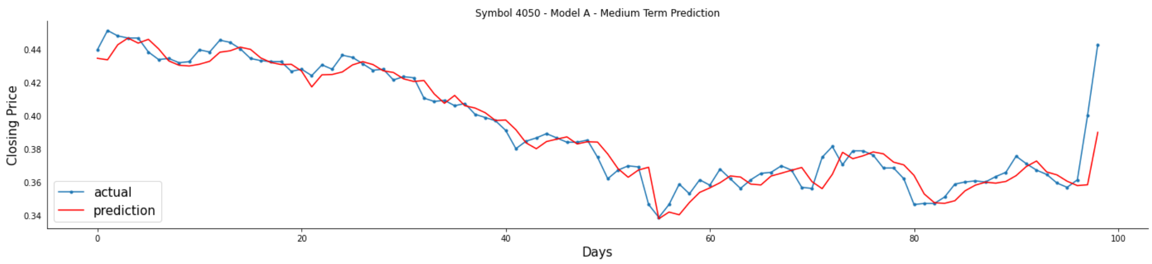


Figure 6.66: Model A - Symbol 4050 - Actual Close versus Predicted Close

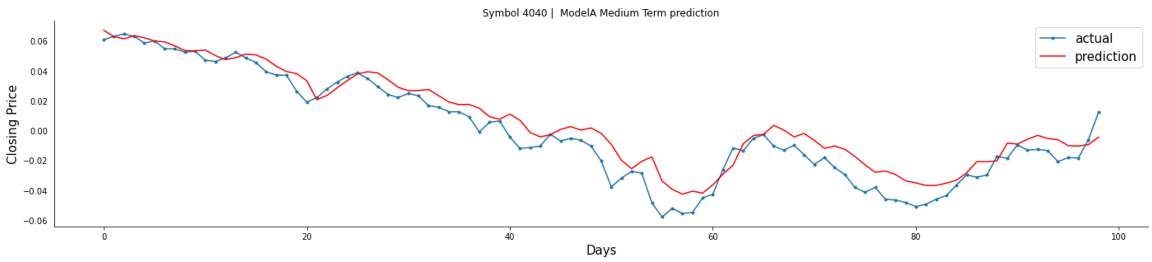


Figure 6.67: Model A - Symbol 4040 - Actual Close versus Predicted Close.

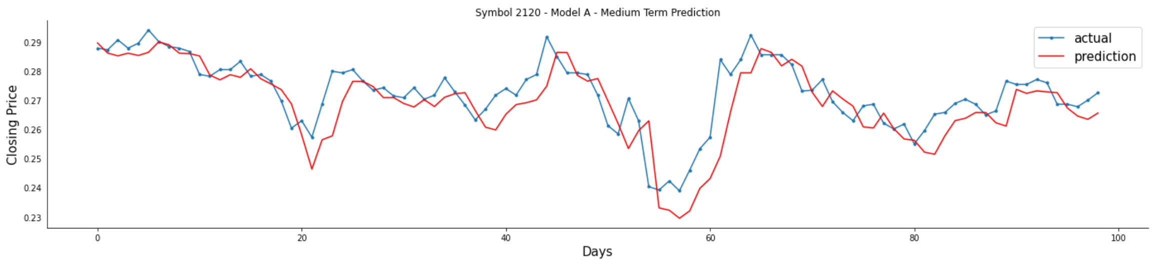


Figure 6.68: Model A - Symbol 2120 - Actual Close versus Predicted Close.

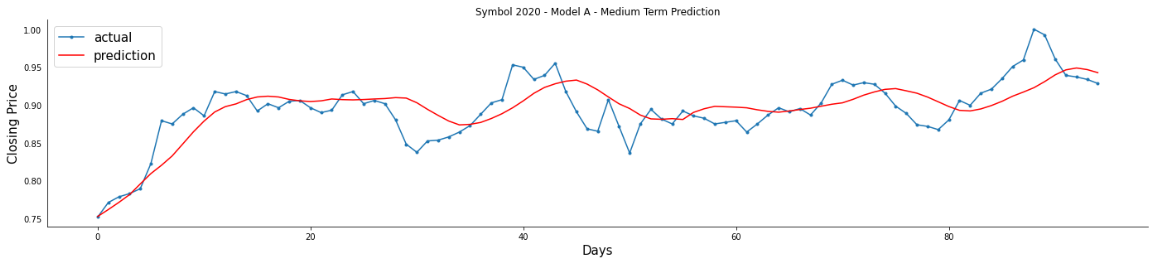


Figure 6.69: Model A - Symbol 2020 - Actual Close versus Predicted Close.

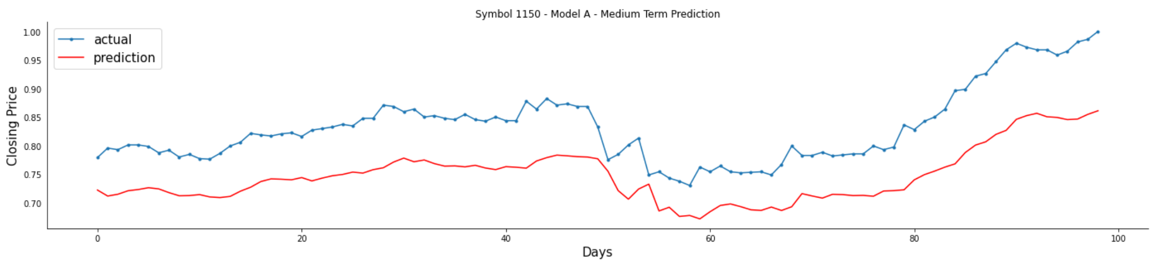


Figure 6.70: Model A - Symbol 1150 - Actual Close versus Predicted Close.

6.3.2.2 Model B Evaluations

Table 6.17 illustrates the Medium Term Mean Absolute Error Percentage of predictions of the selected five stocks with Model B. Mean Absolute Error Percentages are ranging from 2 % to 10% in the model but no stock has registered the best prediction accuracies with Model B which feeds both OHLC and Technical Indicator data to the neural network. Model B generally achieved best accuracies when the network comprised of GRU units.

Table 6.17: Model B – Medium Term Stock Evaluations

Model	**Symbol Name**	**Mean Squared Error**	**Root Mean Squared Error**	**Mean Absolute Error**	**Mean Absolute Percentage Error**
LSTM	4050	0.00052681	0.02319442	0.02050604	5.36141872
GRU	4050	0.00020813	0.00916354	0.00669354	1.72148466
LSTM + GRU	4050	0.00058265	0.01251937	0.00825623	2.14626837
LSTM	4040	0.00009353	0.02876845	0.00838293	3.45363026
GRU	4040	0.00007326	0.00873191	0.00710274	2.37640263
LSTM + GRU	4040	0.00007624	0.00749881	0.00719737	2.45296304
LSTM	2120	0.00051657	0.02276610	0.01993239	7.39879274
GRU	2120	0.00005223	0.00803267	0.00671623	2.50667453

LSTM + GRU	2120	0.00009127	0.01065694	0.00879228	3.23655677
LSTM	2020	0.00158883	0.03887598	0.03119482	3.39930701
GRU	2020	0.00067849	0.02833517	0.02176825	2.94195533
LSTM + GRU	2020	0.00094798	0.03102904	0.02479141	2.79682899
LSTM	1150	0.00104399	0.03440553	0.03043937	3.75775242
GRU	1150	0.00147939	0.04348726	0.03987782	4.89619112
LSTM + GRU	1150	0.001727897	0.039948378	0.032184541	3.707377911

Figure 6.71 to Figure 6.75 visually illustrates the performance of Model B with each Stock Symbol. Figures have shown how Predicted Price from the Model B stacks up with Actual Price of the Stock.

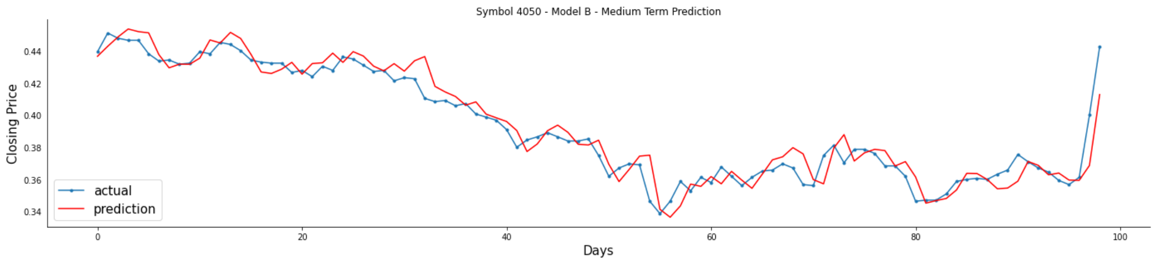


Figure 6.71: Model B - Symbol 4050 - Actual Close versus Predicted Close

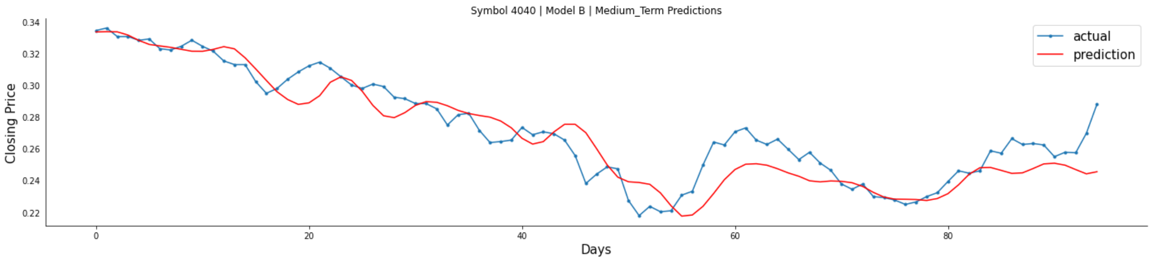


Figure 6.72: Model B - Symbol 4040 - Actual Close versus Predicted Close

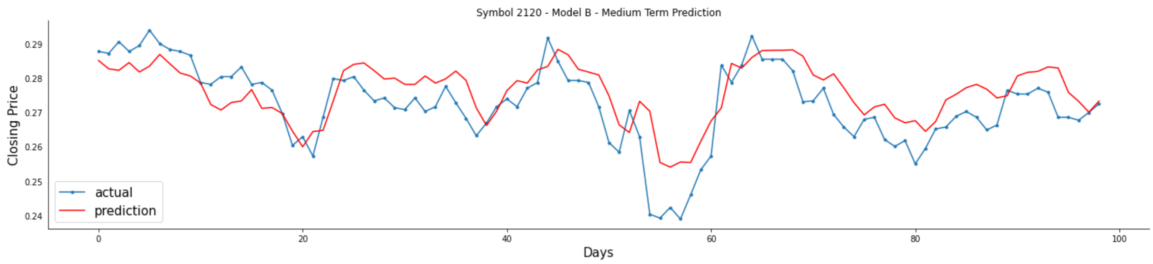


Figure 6.73: Model B - Symbol 2120 - Actual Close versus Predicted Close

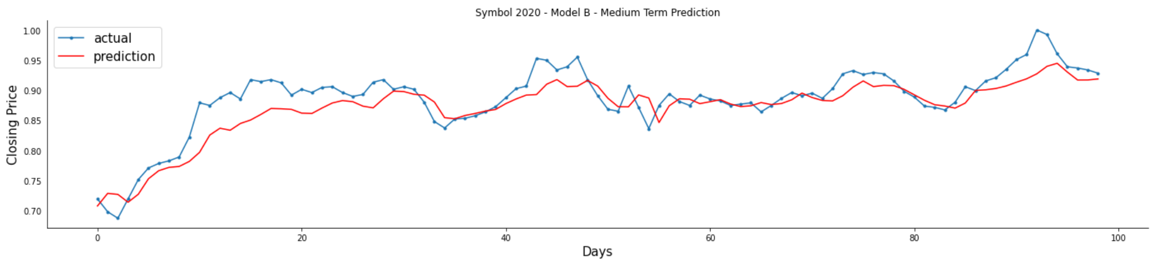


Figure 6.74: Model B - Symbol 2020 - Actual Close versus Predicted Close

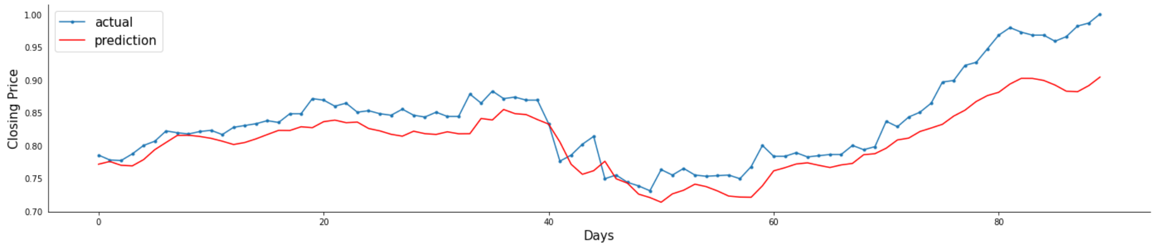


Figure 6.75: Model B - Symbol 1150 - Actual Close versus Predicted Close

6.3.2.3 Model C Evaluations

Table 6.18 illustrates the Medium Term Mean Absolute Error Percentage of predictions of the selected five stocks with Model C. Mean Absolute Error Percentages are ranging from 1 % to 9% in the model but no stock has registered its best prediction accuracies with Model C which feeds both OHLC and Technical Indicator data via a PCA Layer to the neural network. Model C generally achieved best accuracies when the network comprised of both LSTM and GRU units.

Table 6.18: Model C – Medium Term Stock Evaluations

Model	Symbol Name	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	4050	0.00019233	0.01268863	0.01021356	2.66483951
GRU	4050	0.00024590	0.01356928	0.01120581	2.89469361
LSTM + GRU	4050	0.00032011	0.01351919	0.01064511	2.68857646
LSTM	4040	0.00007359	0.00857861	0.00593729	2.96737893
GRU	4040	0.00007193	0.00848100	0.00607393	2.73677839
LSTM + GRU	4040	0.00006352	0.00796997	0.00519278	2.08387278
LSTM	2120	0.00023802	0.01443478	0.01168320	4.29983091
GRU	2120	0.00012694	0.01231171	0.00981557	3.59875035
LSTM + GRU	2120	0.00015719	0.01330790	0.01084170	3.97428513

LSTM	2020	0.00480005	0.07023214	0.06730335	7.53980112
GRU	2020	0.00885624	0.09234136	0.08780344	9.93823624
LSTM + GRU	2020	0.00079225	0.03014163	0.02416295	2.70070076
LSTM	1150	0.00197357	0.04442491	0.02183884	2.78354623
GRU	1150	0.00185382	0.04305604	0.01937973	2.53662373
LSTM + GRU	1150	0.00123514	0.02205005	0.01606607	1.89468944

Figure 6.76 to Figure 6.80 visually illustrates the performance of Model C with each Stock Symbol. Figures have shown how Predicted Price from the Model C stacks up with Actual Price of the Stock. Predicted Price almost mimic the actual price curve indicating a good model performance over the 100 day period.

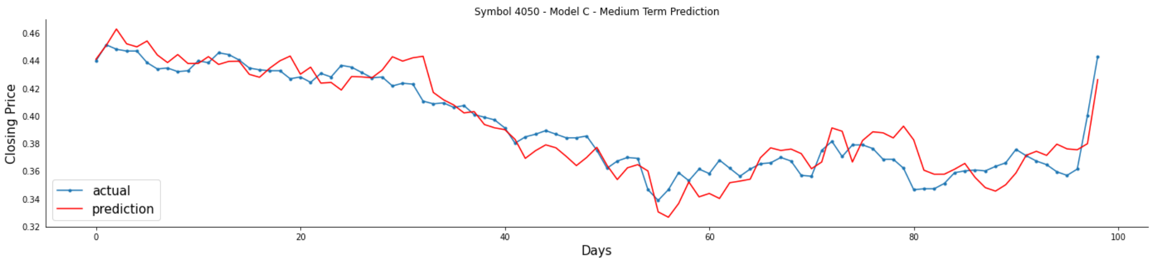


Figure 6.76: Model C - Symbol 4050 - Actual Close versus Predicted Close.

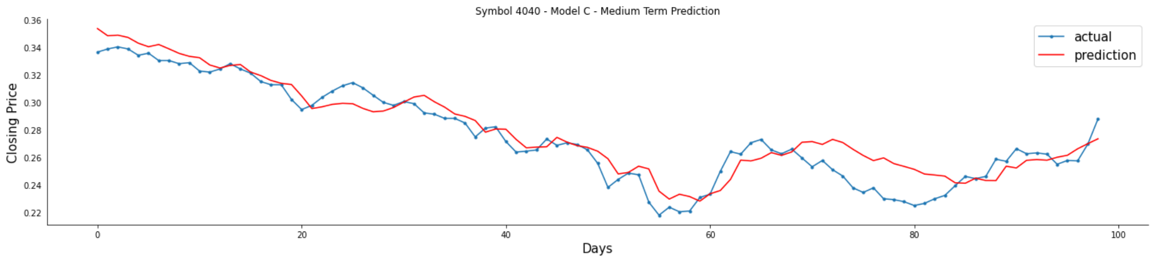
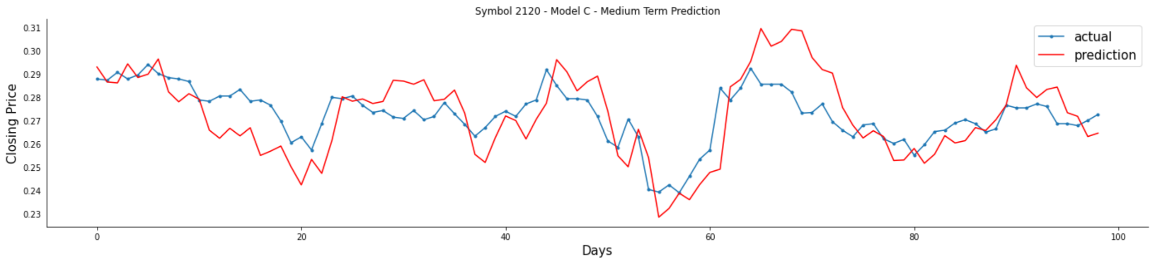


Figure 6.77: Model C - Symbol 4040 - Actual Close versus Predicted Close.



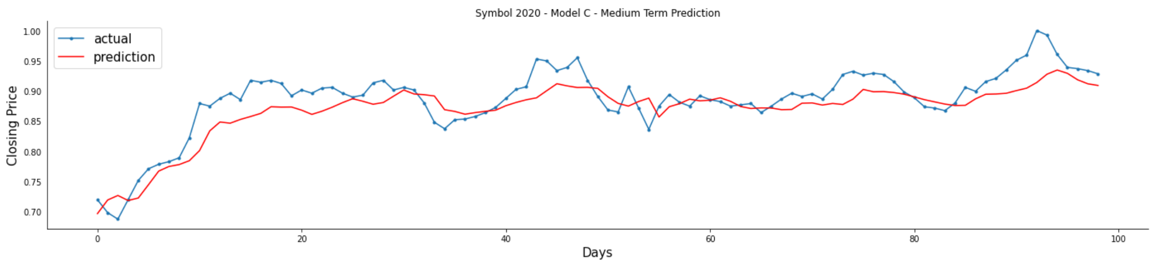


Figure 6.79: Model C - Symbol 2020 - Actual Close versus Predicted Close.

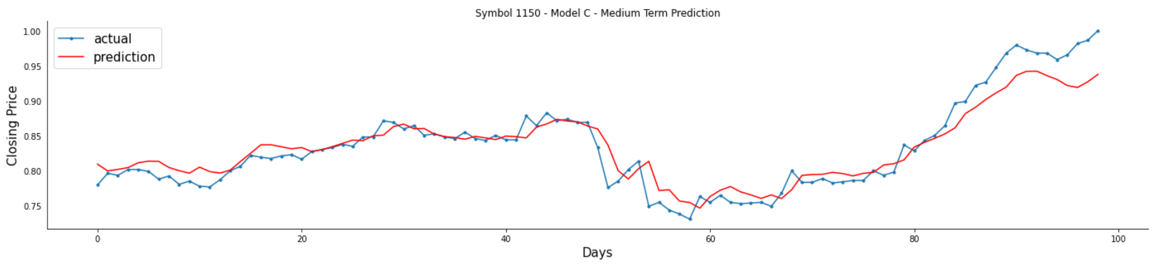


Figure 6.80: Model C - Symbol 1150 - Actual Close versus Predicted Close.

6.3.2.4 Model D Evaluations

Table 6.19 illustrates the Medium Term Mean Absolute Error Percentage of predictions of the selected five stocks with Model D. Mean Absolute Error Percentages are ranging from 1 % to 3% in the model. Every Stock except stock 2120 has registered its best prediction accuracies with Model D .It is the model which feeds both OHLC data directly and Technical Indicator data via a PCA Layer to the neural network. Model D generally achieved best accuracies when the network comprised of both LSTM and GRU units. Model D was the best model emerged for medium term predictions in stocks.

Table 6.19: Model D – Medium Term Stock Evaluations

Model	Symbol Name	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	4050	0.00022155	0.01233083	0.01032926	2.63772607
GRU	4050	0.00033879	0.01104864	0.00833616	2.12551212
LSTM + GRU	**4050**	**0.00041882**	**0.00970862**	**0.00611020**	**1.56723225**
LSTM	4040	0.00005367	0.00732614	0.00473774	1.97378378
GRU	4040	0.00006820	0.00825855	0.00538299	2.16397878
LSTM + GRU	**4040**	**0.00005183**	**0.02276585**	**0.00453794**	**1.95039297**
LSTM	2120	0.00010721	0.01156839	0.00935159	3.41944408
GRU	2120	0.00087746	0.02856092	0.02660947	9.79390430

LSTM + GRU	2120	0.00016983	0.01408989	0.01141578	4.20044661
LSTM	2020	0.00107029	0.03309470	0.02841215	3.19455767
GRU	2020	0.00112900	0.03371144	0.02596649	2.86428213
LSTM + GRU	**2020**	**0.00069174**	**0.02630100**	**0.01836647**	**2.09048729**
LSTM	1150	0.00247676	0.04703911	0.03279988	3.70269752
GRU	1150	0.01698292	0.10317054	0.09471014	11.08248043
LSTM + GRU	**1150**	**0.00098669**	**0.03141165**	**0.01985347**	**1.71436114**

Figure 6.81 to Figure 6.85 visually illustrates the performance of Model D with each Stock Symbol. Figures have shown how Predicted Price from the Model D stacks up with Actual Price of the Stock. Predicted Price is very similar to the actual price curve indicating a good model performance over the 100 day period.

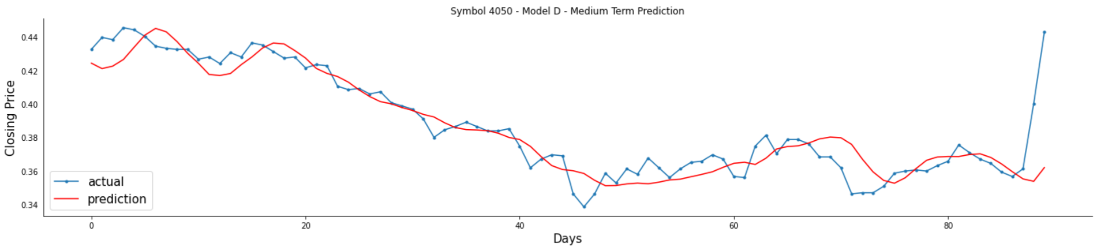


Figure 6.81: Model D - Symbol 4050 - Actual Close versus Predicted Close.

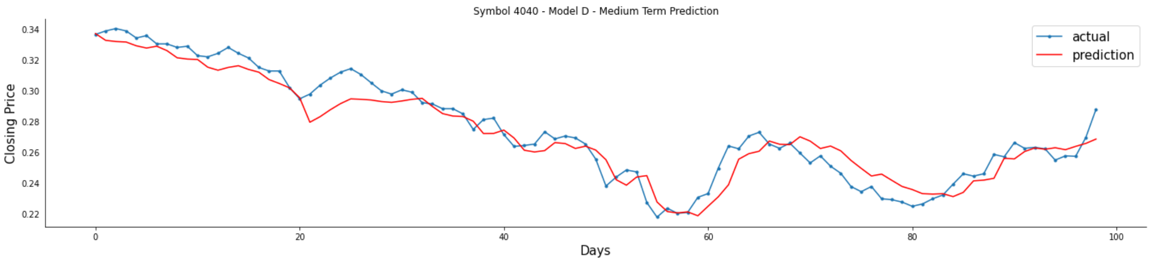


Figure 6.82: Model D - Symbol 4040 - Actual Close versus Predicted Close.

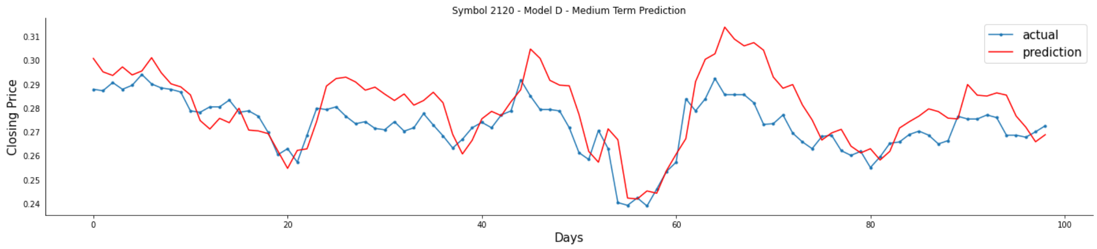


Figure 6.83: Model D - Symbol 2120 - Actual Close versus Predicted Close.

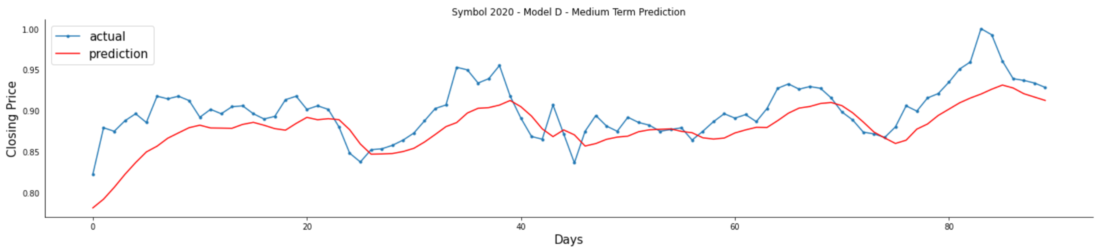


Figure 6.84: Model D - Symbol 2020 - Actual Close versus Predicted Close.

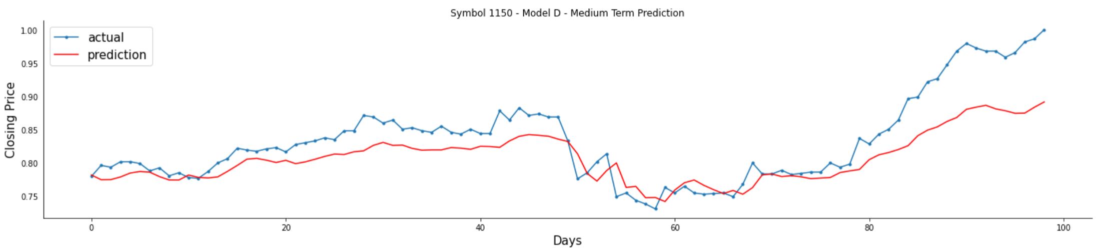


Figure 6.85: Model D - Symbol 1150 - Actual Close versus Predicted Close.

6.4 Long Term Evaluation and Analysis

Long Term evaluation is defined as predicting the close price behavior for next 500 day period. It is primarily important for long term investors who anticipate returns in the long term. It can be used as a predictor of a country's economy over the long term. Micro-economic quantitative analysis do explain a fair share of the price behavior in long term. Unforeseen activities do have a less impact to the models as the model captures the variations over time and kind of generalize the behavior.

6.4.1 Tadawul All Share Index (TASI) Evaluations

Prediction was done for Tadawul All Share Index commonly known as TASI. Results of the benchmark models for the Long Term are as follows.

Table 6.19 TASI Index – Long Term Results of the Bench Mark Models

Model	**MSE**	**RMSE**	**MAE**	**MAPE**
Moving Average	0.0000592842	0.0076996264	0.0057364140	1.6414535254
Exponential Moving Average	**0.0000547962**	**0.0074024470**	**0.0054662739**	**1.56578409**
ARMA	0.0001778280	0.0421696573	0.0082235218	1.9714662391
GARCH	3.0601127206	1.7493177872	1.2508288858	8.2839541826
SVM– OHLC Only	0.0012100354	0.0347855624	0.0335276009	9.68096247
SVM- OHLC and TI	0.0032593813	0.0570909916	0.0564219368	16.01295950
FFNN	0.0001001744	0.0100087151	0.0069549568	5.5472631454

Exponential Moving Average which is a statistical method recorded the best overall Mean Absolute Error Percentage value in benchmark models. A variable moving average window was tried and best results were obtained for rolling window size of three.

6.4.1.1 Model A Evaluations

6.4.1.1.1 Optimizing the Window Size

Window size optimization was performed by keeping all other parameters a constant value. Mean Squared Error with the variation of Window Size was observed to obtain a precise window size in days. Figure 6.86 shows that a window size of 1 achieved the lowest error rate. It meant that tomorrow close price is heavily dependent on today's close price in Model A where OHLC data is directly fed to the neural network.

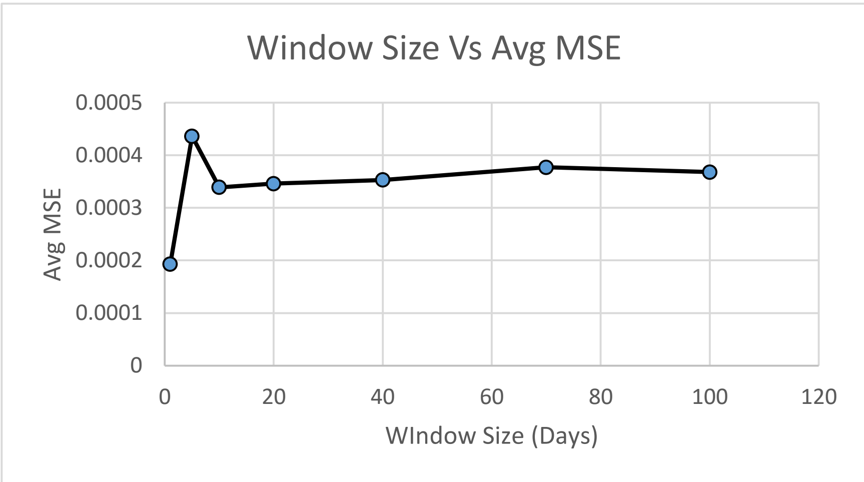


Figure 6.86: Model A –Long Term Window Size vs. MSE Variation

6.4.1.1.2 Optimizing the Neuron Count in Each Layer

Neuron Count in each layer was also tuned by keeping all other parameters constant and running the test 10 separate times to record the average Mean Square Error performance. Figure 6.87 show the variation between number of neurons per layer and MSE. It clearly indicates lower error rates when the neuron count ranges from 10 to 20 per layer.

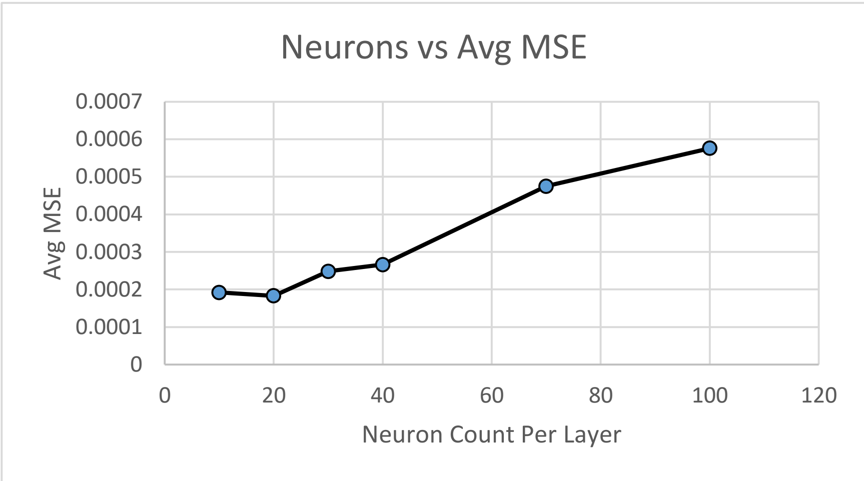


Figure 6.87: Model A –Long Term Neuron Count vs. MSE Variation

6.4.1.1.3 Optimizing the training Batch Size of the Network

Optimum Batch Size was found by keeping all other parameters in a constant value. Batch Size search was done following a grid search methodology. Each batch size point was evaluated repeatedly 10 times to obtain the averaged Mean Squared Error. Figure 6.88 shows the variation between Batch Size and MSE. It indicates better error rates for batch size of 16 for Model A where OHLC and Technical Indicator data are fed directly to the neural network.

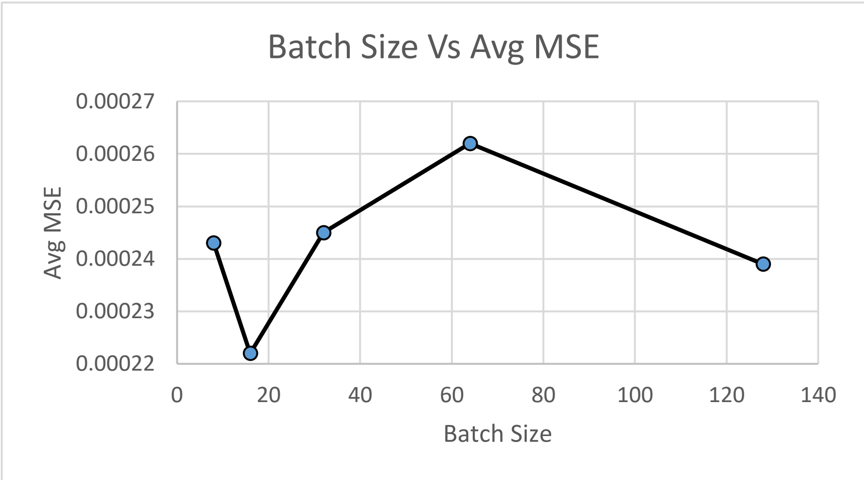


Figure 6.88: Model A –Long Term Batch Size vs. MSE Variation

6.4.1.1.4 Deep Neural Network Layer Count Tuning

Each model is trained and tested with All LSTM layers, All GRU layers and Hybrid of LSTM and GRU layer. Hence I tested with Mean Squared Error variation with LSTM and GRU Layer count in order to get a clear idea of the sweet spot which I could have the best performance. Figure 6.89 and Figure 6.90 shows the variation of Mean Squared Error with No of LSTM and GRU Layer count. Best Performance for both LSTM and GRU networks were given when the layer count is between 1 and 3.

LSTM Layer Count versus Mean Squared Error Variation

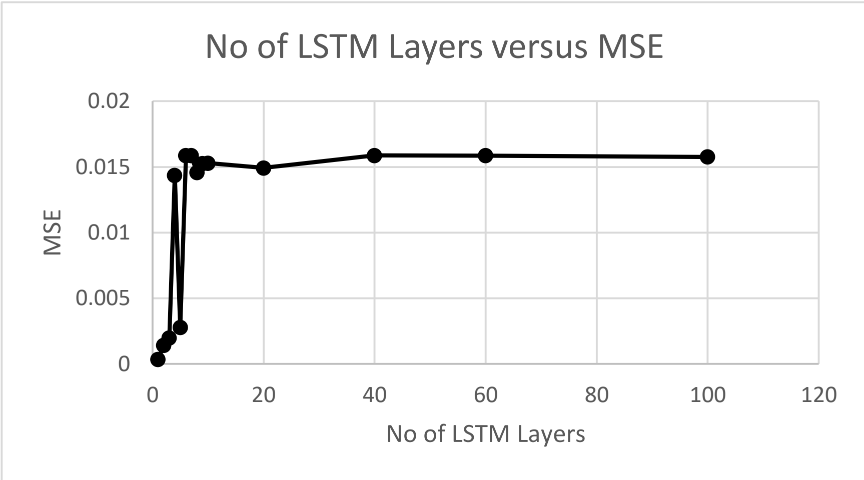


Figure 6.89: Model A –Long Term LSTM Layer Count vs. MSE Variation

GRU Layer Count versus Mean Squared Error Variation

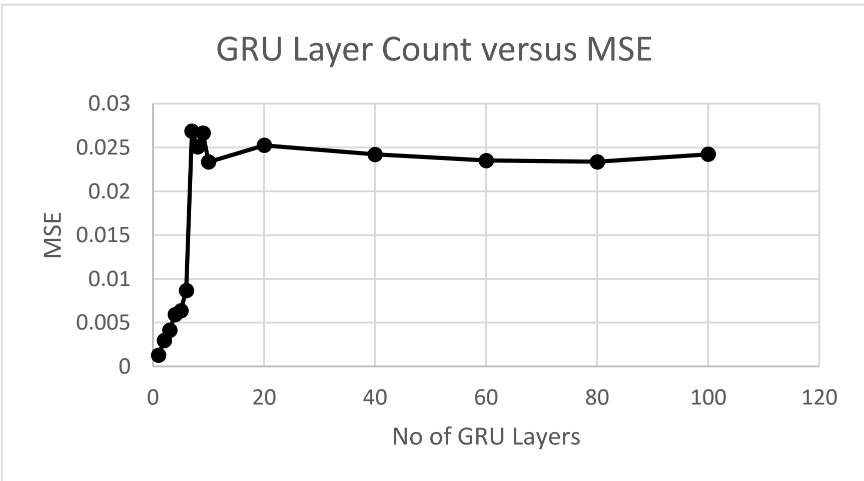


Figure 6.90: Model A –Long Term GRU Layer Count vs. MSE Variation

6.4.1.1.5 Overall Model Performance

Model which feeds OHLC data directly to the Neural Network (Model A) is trained with all LSTM, All GRU and Hybrid network with LSTM and GRU Layers. Table 6.20 tabulates the best results of each network. Hybrid Network which comprised of LSTM and GRU Layers was giving the best performance in terms of Maximum Absolute Percentage Error. MAPE of 3.93% was the best Error rate of the Model A. Figure 6.91 shows how Actual Close price varies with predicted close price over a period of 500 days. It was expected that error rate would be much higher in longer term than that of short and medium terms due to the extended prediction period of 500 days. Hence an error percentage of 3.93% was considered good as it did not deviate hugely from medium term percentages of 1-2%.

Table 6.20 TASI Index – Long Term Results of Model A

Model	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	0.0000472570	0.0068743723	0.0054944959	4.2391114235
GRU	**0.0000399529**	**0.0063208295**	**0.0049511851**	**3.9346888065**
LSTM + GRU	0.0001131230	0.0106359301	0.0091741392	6.4027786255

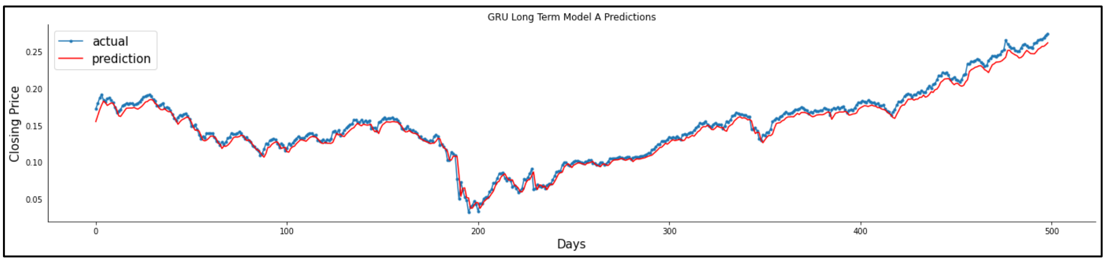


Figure 6.91: Model A Long Term - TASI Actual Close vs. Predicted Close.

6.4.1.2 Model B Evaluations

6.4.1.2.1 Optimizing the Window Size

All other parameters were kept a constant value and observed the behavior of the Mean Squared Error with the variation of Window Size to obtain a precise window size in days. The experiment is performed 10 times and Mean Squared Error is averaged out at the end. Figure 6.92 shows the variation of Window Size with MSE. Lowest error rates were indicated when Window Size is 1 day. It meant that tomorrow closing price tend to depend only on closing price of previous day.

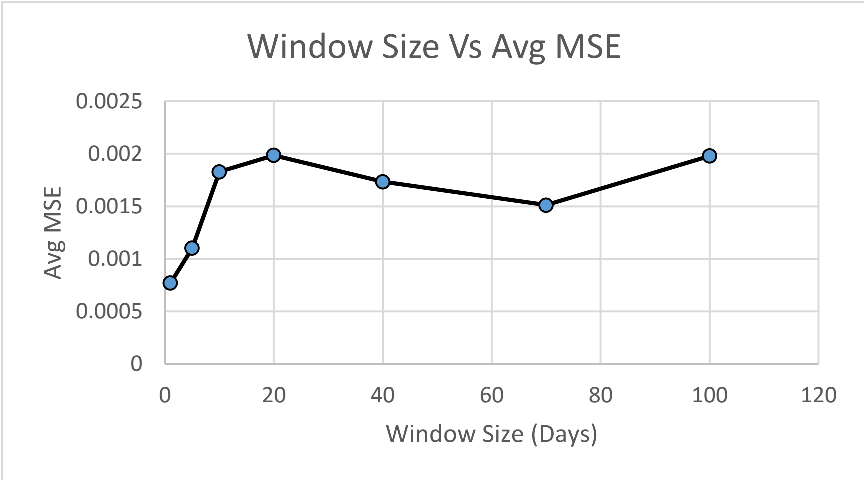


Figure 6.92: Model B - Long Term - Window Size vs. MSE Variation

6.4.1.2.2 Optimizing the Neuron Count in Each Layer

Neuron Count in each layer was also tuned by keeping all other parameters constant and running the test 10 separate times to record the average performance. Figure 6.93 show the variation between number of neurons per layer and MSE. It clearly indicates lower error rates when the neuron count is 40 neurons per layer.

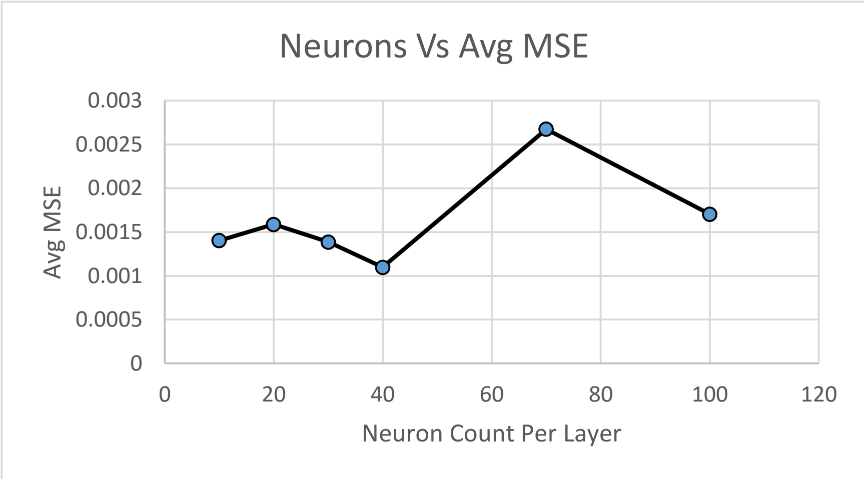


Figure 6.93: Model B –Long Term - Neuron Count vs. MSE Variation

6.4.1.2.3 Optimizing the training Batch Size of the Network

Optimum Batch Size was found by keeping all other parameters in a constant value. Batch Size search was done following a grid search methodology. Each batch size point was evaluated repeatedly 10 times to obtain the averaged Mean Squared Error. Figure 6.94 shows the variation between Batch Size and MSE. It indicates better error rates for batch size of 64 for Model B where OHLC and Technical Indicator data are fed directly to the neural network.

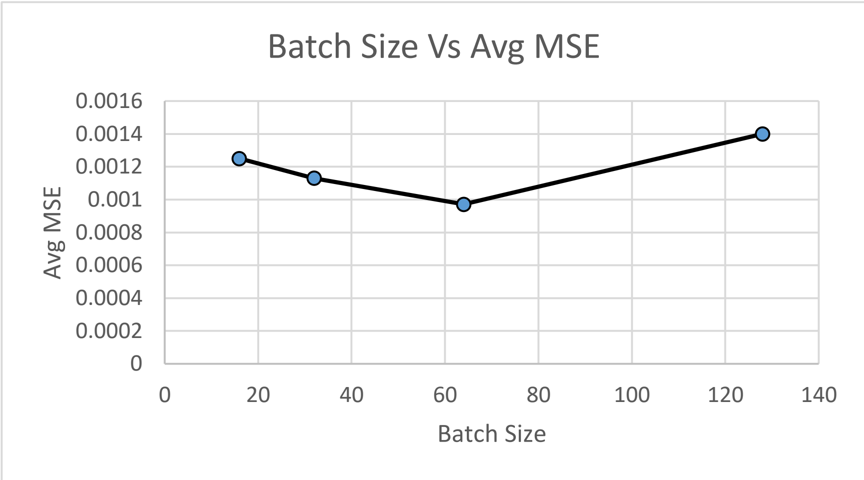


Figure 6.94: Model B – Long Term - Batch Size vs. MSE Variation

6.4.1.2.4 Deep Neural Network Layer Count Tuning

Model which feeds OHLC and Technical Indicator data directly to the Neural Network (Model B) is trained and tested with All LSTM network and All GRU network. Figure 6.95 and Figure 6.96 shows the variation of Mean Squared Error with No of LSTM and GRU Layer count. Best Performance for both LSTM and GRU networks were given when the layer count is between 1 and 5. Any number of layers above 5 showed very high error rates.

<u>LSTM Layer Count versus Mean Squared Error Variation</u>

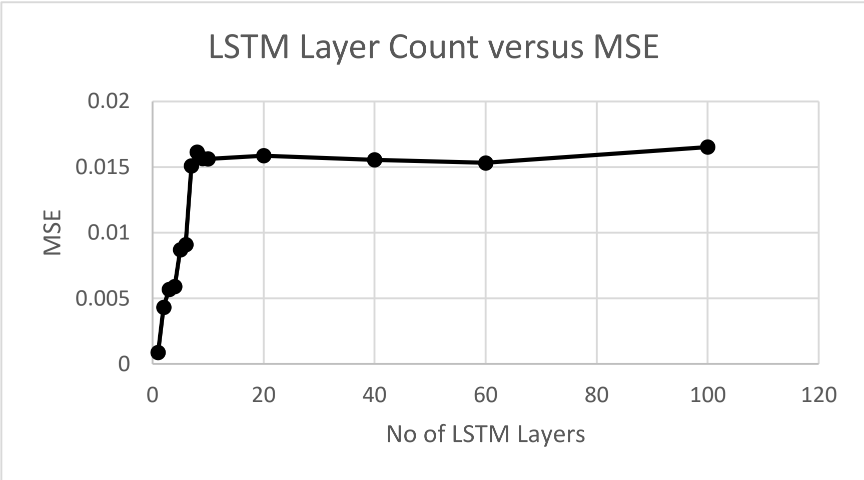


Figure 6.95: Model B –Long Term LSTM Layer Count vs. MSE Variation

<u>GRU Layer Count versus Mean Squared Error Variation</u>

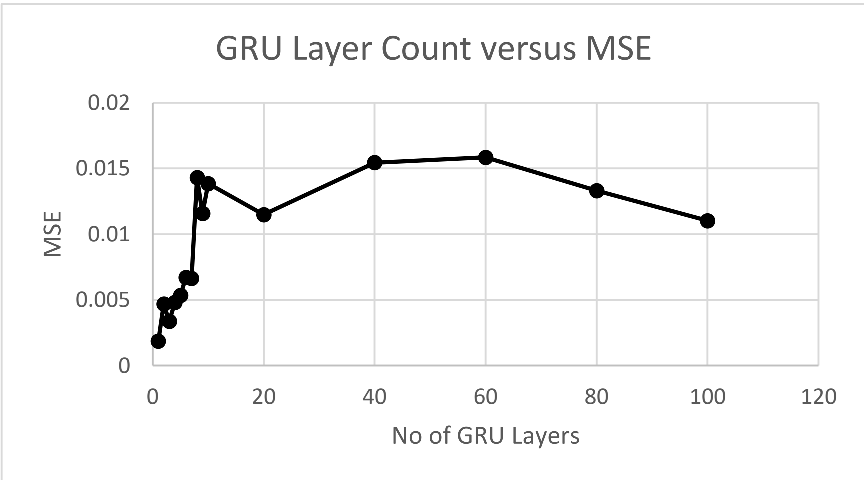


Figure 6.96: Model B –Long Term - GRU Layer Count vs. MSE Variation

6.4.1.2.5 Overall Model Performance

Model which feeds OHLC and Technical Indicator data directly to the Neural Network (Model B) is trained with all LSTM, All GRU and Hybrid network with LSTM and GRU Layers. Table 6.21 tabulates the best results of each network. Hybrid Network which comprised of LSTM and GRU Layers gave the best performance in terms of Maximum Absolute Percentage Error. MAPE of 3.91% was the best figure which could be obtained via Model B. It was the same error percentage as previous Model A. Therefore no huge performance gap was observed between two models. Figure 6.97 shows how Actual Close price varies with predicted close price from Model B over a period of 500 days.

Table 6.21 TASI Index – Long Term Results of Model B

Model	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	0.0000506978	0.0071202419	0.0053811772	4.4955010414
GRU	**0.0000355713**	**0.0059641660**	**0.0044470159**	**3.9515802860**
LSTM + GRU	0.0001100444	0.0104902070	0.0081424965	6.4196219444

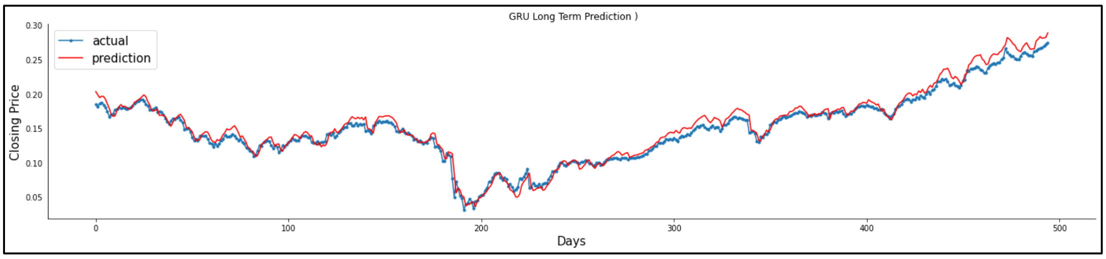


Figure 6.97: Model B Long Term - TASI Actual Close vs. Predicted Close.

6.4.1.3 Model C Evaluations

6.4.1.3.1 Optimizing the Window Size

All other parameters were kept a constant value and observed the behavior of the Mean Squared Error with the variation of Window Size to obtain a precise window size in days. The experiment is performed 10 times and Mean Squared Error is averaged out at the end. Figure 6.98 shows the variation of Window Size with MSE. Lowest error rates were indicated when Window Size is 1 day. It meant that tomorrow closing price tend to depend on closing price of previous day only.

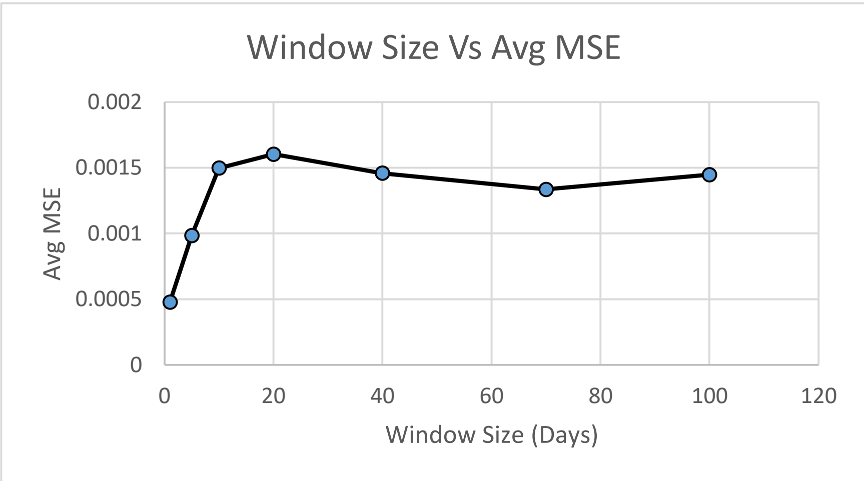


Figure 6.98: Model C - Long Term - Window Size vs. MSE Variation

6.4.1.3.2 Optimizing the Neuron Count in Each Layer

Neuron Count in each layer was also tuned by keeping all other parameters constant and running the test 10 separate times to record the average performance. Figure 6.99 show the variation between number of neurons per layer and MSE. It clearly indicates lower error rates when the neuron count is between 20 and 40 neurons per layer.

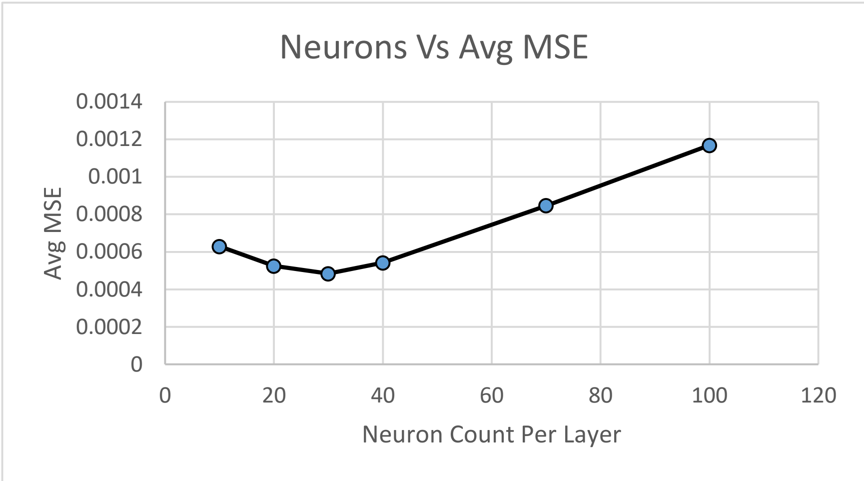


Figure 6.99: Model C –Long Term - Neuron Count vs. MSE Variation

6.4.1.2.3 Optimizing the training Batch Size of the Network

Optimum Batch Size was found by keeping all other parameters in a constant value. Batch Size search was done following a grid search methodology. Each batch size point was evaluated repeatedly 10 times to obtain the averaged Mean Squared Error. Figure 6.100 shows the variation between Batch Size and MSE. It indicates better error rates for batch size of 16 for Model C where OHLC and Technical Indicator data are fed via PCA Layer to the neural network.

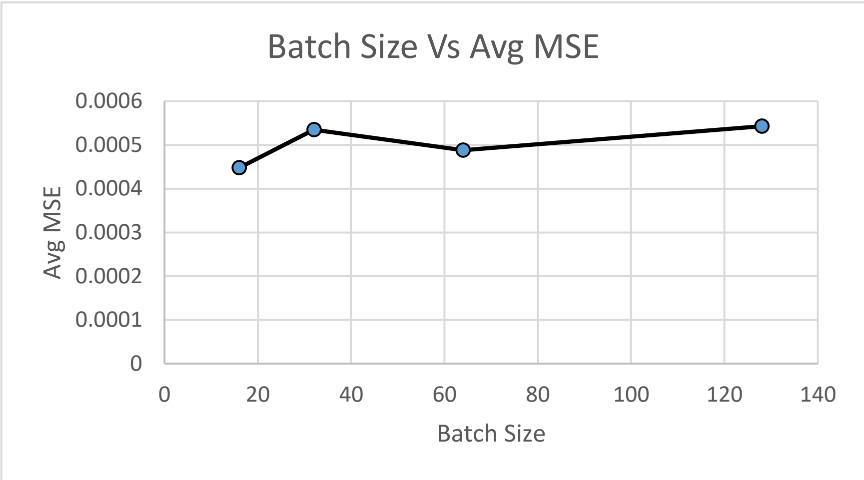


Figure 6.100: Model C – Long Term - Batch Size vs. MSE Variation

6.4.1.3.4 Deep Neural Network Layer Count Tuning

Model which feeds OHLC and Technical Indicator data via PCA Layer to the Neural Network (Model C) is trained and tested with All LSTM network and All GRU network. Figure 6.101 and Figure 6.102 shows the variation of Mean Squared Error with No of LSTM and GRU Layer count. Best Performance in GRU networks were given when the layer count is between 1 and 3 and 1 to 4 neurons per layer in LSTM Layers. Any number of layers above 5 showed very high error rates.

LSTM Layer Count versus Mean Squared Error Variation

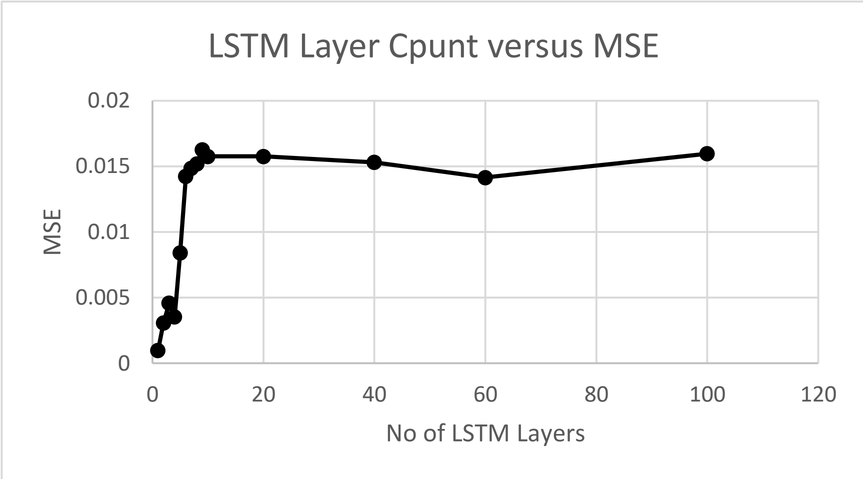


Figure 6.101: Model C –Long Term LSTM Layer Count vs. MSE Variation

GRU Layer Count versus Mean Squared Error Variation

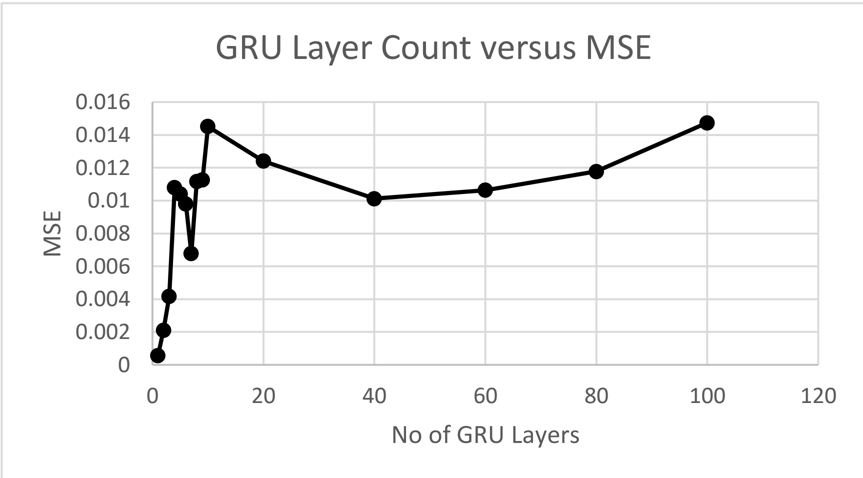


Figure 6.101: Model C –Long Term - GRU Layer Count vs. MSE Variation

6.4.1.3.5 Overall Model Performance

Model which feeds OHLC and Technical Indicator data via PCA Layer to the Neural Network (Model C) is trained with all LSTM, All GRU and a Hybrid network with LSTM and GRU Layers. Table 6.22 tabulates the best results of each network. Hybrid Network which comprised of LSTM and GRU Layers was giving the best performance in terms of Maximum Absolute Percentage Error. MAPE of 2.14% was the best error percentage achieved from the model in long term predictions. Model C showed a decrement of MAPE from 3% to 2% from the previous two models. Figure 6.102 shows how Actual Close price varies with predicted close price calculated from Model C over a period of 100 days.

Table 6.22 TASI Index – Long Term Results of Model C

Model	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	0.00010668798	0.01032898761	0.00855608098	2.33598184585
GRU	0.00012026097	0.01096635684	0.00895087979	2.51229834556
LSTM + GRU	**0.00008932926**	**0.00945141539**	**0.00781683623**	**2.14929127693**

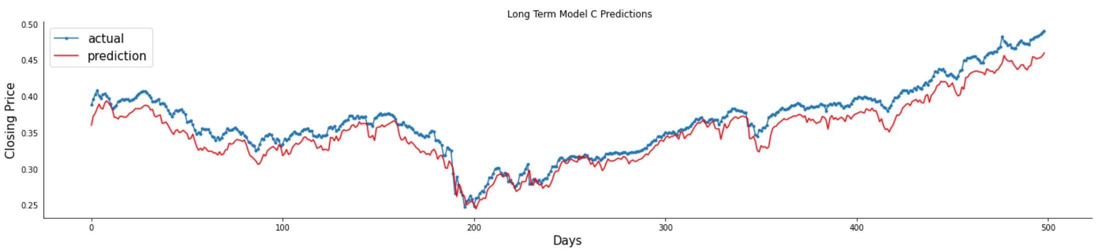


Figure 6.102: Model C Long Term - TASI Actual Close vs. Predicted Close.

6.4.1.4 Model D Evaluations

6.4.1.4.1 Optimizing the Window Size

All other parameters were kept a constant value and observed the behavior of the Mean Squared Error with the variation of Window Size to obtain a precise window size in days. The experiment is performed 10 times and Mean Squared Error is averaged out at the end. Figure 6.103 shows the variation of Window Size with MSE. Lowest error rates were indicated when Window Size is 1 and 20 days. It means model is working with high accuracy when it is depending on just previous days' close price as well as close price of 20 previous days as well.

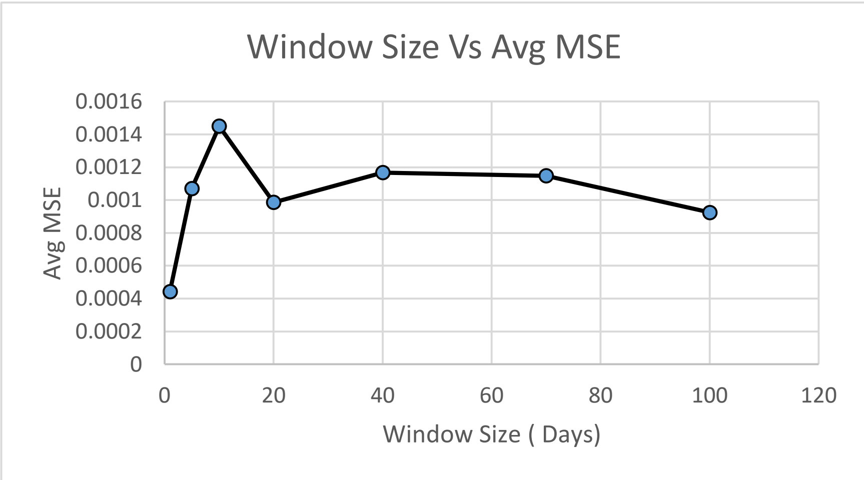


Figure 6.103: Model D - Long Term - Window Size vs. MSE Variation

6.4.1.4.2 Optimizing the Neuron Count in Each Layer

Neuron Count in each layer was also tuned by keeping all other parameters constant and running the test 10 separate times to record the average performance. Figure 6.104 show the variation between number of neurons per layer and MSE. It clearly indicates lower error rates when the neuron count is in the range of 20 to 30 neurons per layer.

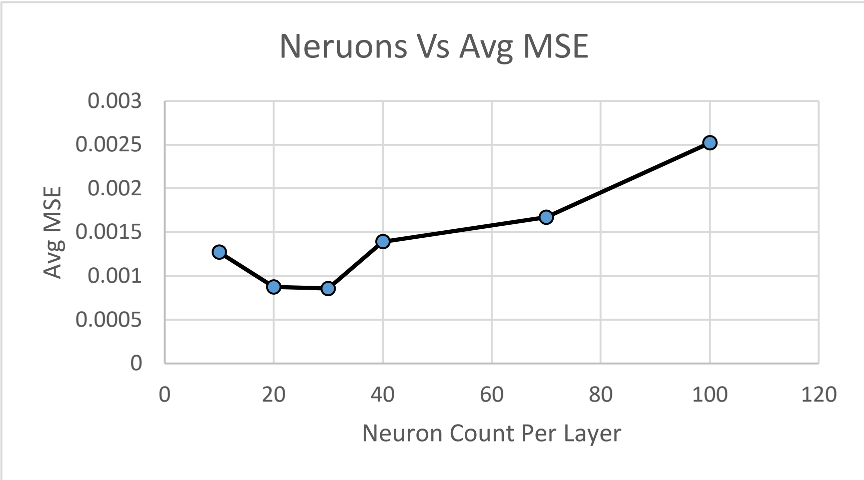


Figure 6.104: Model D –Long Term - Neuron Count vs. MSE Variation

6.4.1.4.3 Optimizing the training Batch Size of the Network

Optimum Batch Size was found by keeping all other parameters in a constant value. Batch Size search was done following a grid search methodology. Each batch size point was evaluated repeatedly 10 times to obtain the averaged Mean Squared Error. Figure 6.104 shows the variation between Batch Size and MSE. It indicates better error rates for batch size of 32 for Model D where OHLC data is fed directly and Technical Indicator data is fed via PCA Layer to the neural network.

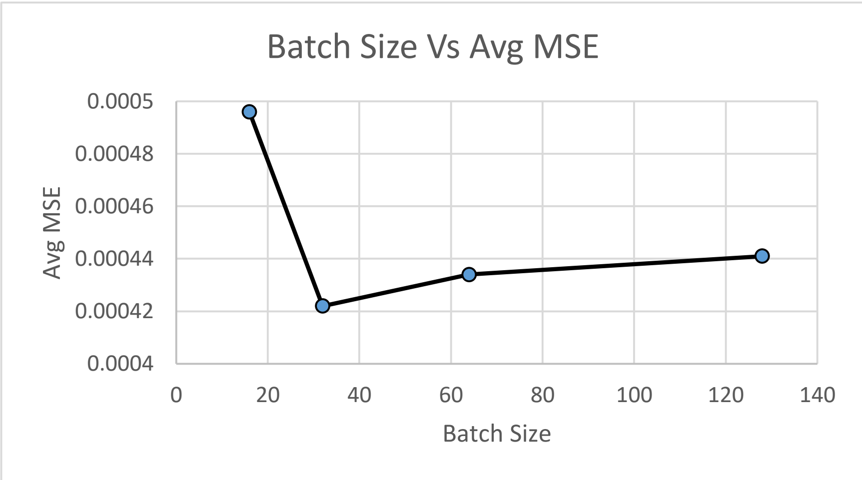


Figure 6.104: Model D – Long Term - Batch Size vs. MSE Variation

6.4.1.4.4 Deep Neural Network Layer Count Tuning

Model which feeds OHLC data directly and Technical Indicator data via PCA Layer to the Neural Network (Model D) is trained and tested with All LSTM network and All GRU network. Figure 6.105 and Figure 6.106 shows the variation of Mean Squared Error with No of LSTM and GRU Layer count. Best Performance in both LSTM and GRU networks were given when the layer count is between 1 and 4. Any number of layers above 5 showed very high error rates.

<u>LSTM Layer Count versus Mean Squared Error Variation</u>

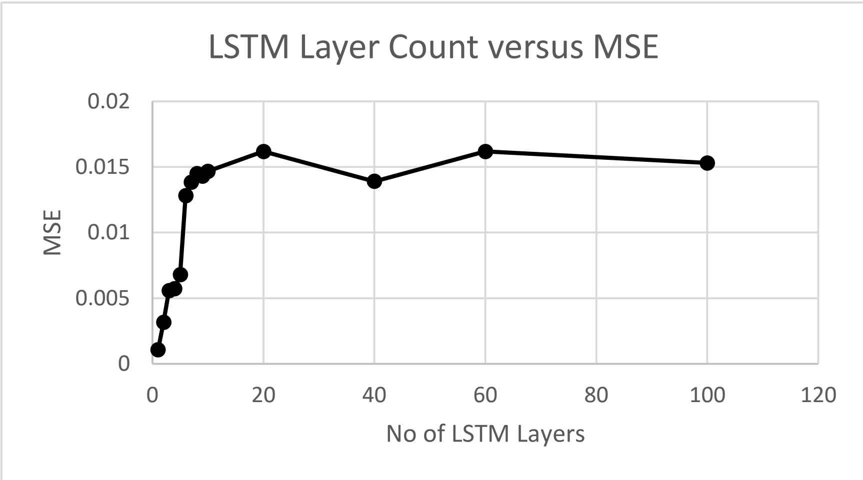


Figure 6.105: Model D –Long Term LSTM Layer Count vs. MSE Variation

<u>GRU Layer Count versus Mean Squared Error Variation</u>

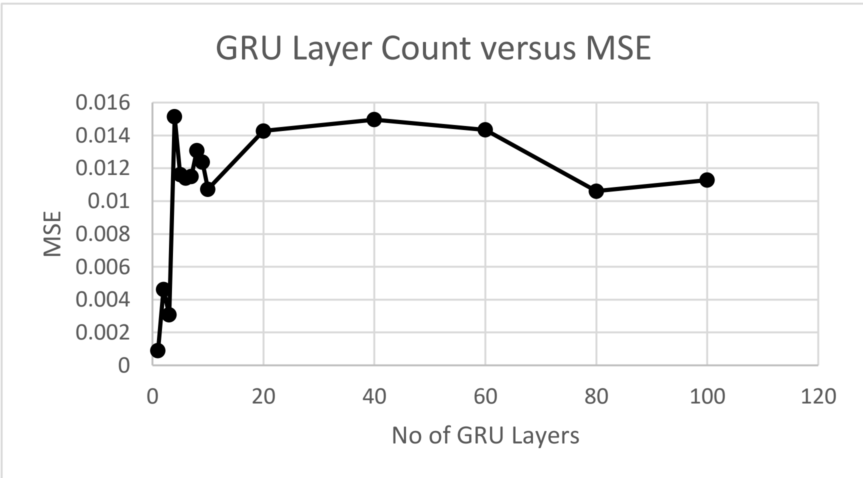


Figure 6.106: Model D –Long Term - GRU Layer Count vs. MSE Variation

6.4.1.4.5 Overall Model Performance

Model which feeds OHLC data directly and Technical Indicator data via PCA Layer to the Neural Network (Model D) is trained with all LSTM, all GRU and a Hybrid network with LSTM and GRU Layers. Table 6.23 tabulates the best results of each network. Hybrid Network which comprised of LSTM and GRU Layers was giving the best performance in terms of Maximum Absolute Percentage Error. MAPE of 1.53% was the best error percentage achieved from the model in long term predictions. Model D showed a decrement of MAPE from 2% to 1.5% from the previous Model C. So this model can be cited as the best model for Long Term Predictions. Figure 6.107 shows how Actual Close price varies with predicted close price calculated from Model C over a period of 500 days.

Table 6.23 TASI Index – Long Term Results of Model D

Model	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	0.00007261651	0.008521532	0.006322651	1.815066933
GRU	0.00007238049	0.008507672	0.006921686	1.919279694
LSTM + GRU	**0.00004693101**	**0.00706618838**	**0.0053777787**	**1.5364066362**

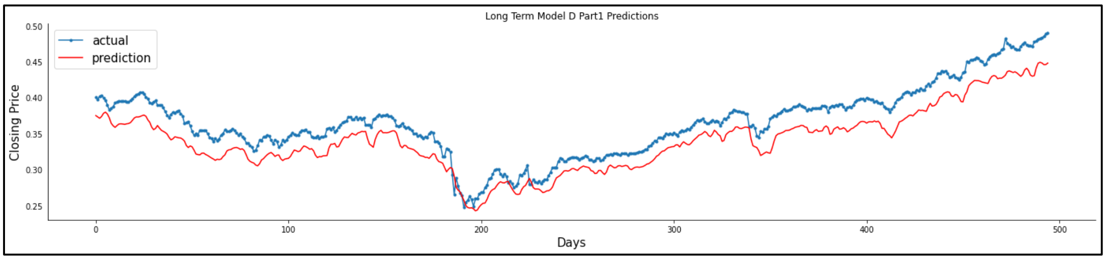


Figure 6.107: Model D Long Term - Actual Close vs. Predicted Close

6.4.2 Stock Evaluations

All four models were trained with available data for each symbol and prediction was given on its closing price for next 500 days as Long term predictions. Results are stacked as below with each model. Charts have been attached with predicted vs. actual which gives a good visual comparison apart from quantitative analysis.

6.4.2.1 Model A Evaluations

Performance of the Five Symbols in Long Term are as follows. Long Term Prediction is used to predict the close price for the next 500 days into the future. . Best error rate of 2.35% is achieved for Symbol 4040. Overall Error Rates are ranging from 2% to 15%. Higher Error Percentages should be anticipated as the prediction period now is 500 days not 25 or 100 days and the error rates are also expected to shoot up.

Table 6.24: Model A – Long Term Stock Evaluations

Model	**Symbol Name**	**Mean Squared Error**	**Root Mean Squared Error**	**Mean Absolute Error**	**Mean Absolute Percentage Error**
LSTM	4050	0.00014756	0.01191580	0.00918450	2.48088002
GRU	4050	0.00101069	0.03167788	0.03026000	8.09641171
LSTM + GRU	4050	0.00032319	0.01788287	0.01502433	3.83366799
LSTM	4040	0.00005018	0.00708397	0.00621939	2.35293793
GRU	4040	0.00005674	0.00753249	0.00667826	2.68329994
LSTM + GRU	4040	0.00006633	0.00814425	0.00736489	2.98673673

LSTM	2120	0.00014802	0.01211004	0.01012452	7.24755573
GRU	2120	0.00271631	0.05168426	0.04715057	32.07691574
LSTM + GRU	2120	0.00066864	0.02574950	0.02351442	17.03754616
LSTM	2020	0.00145722	0.03643870	0.02551998	4.00034189
GRU	2020	0.00060547	0.02437119	0.02176275	4.64310598
LSTM + GRU	2020	0.00170529	0.04129517	0.03758626	7.70318937
LSTM	1150	0.00145908	0.03607273	0.02868782	4.71516848
GRU	1150	0.00377987	0.05817764	0.04734208	7.70453691
LSTM + GRU	1150	0.01670657	0.12264185	0.09882429	15.76468849

Figure 6.108 to Figure 6.112 visually illustrates the performance of Model A with each Stock Symbol. Figures have shown how Predicted Price from the Model A stacks up with Actual Price of the Stock. The plot of Symbol 4040 achieved the highest long term performance with Model A. It shows much closer predicted curve to the actual curve.

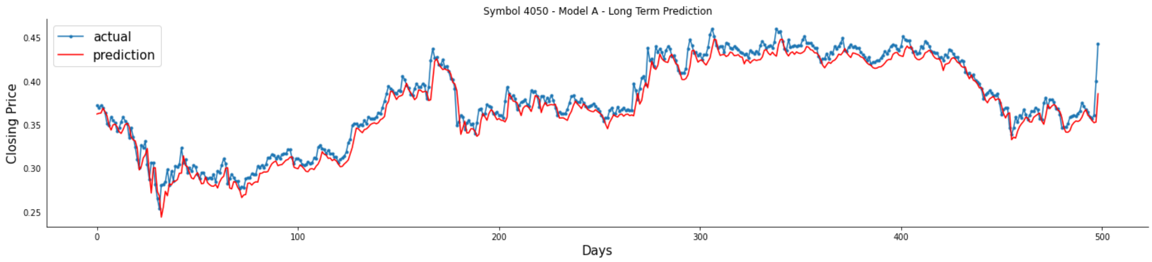


Figure 6.108: Model A - Symbol 4050 - Actual Close versus Predicted Close.

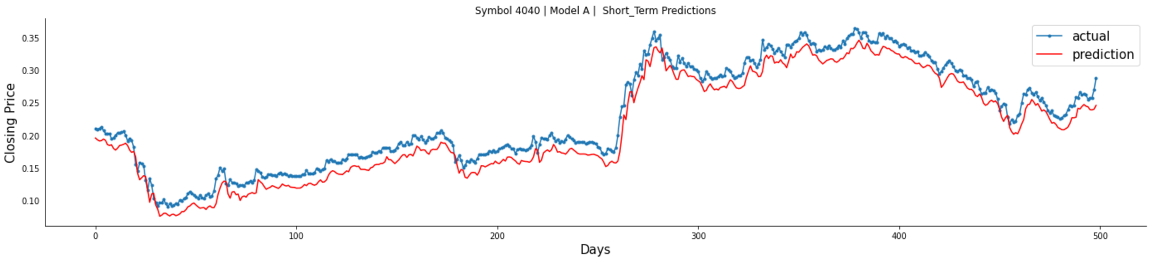


Figure 6.109: Model A - Symbol 4040 - Actual Close versus Predicted Close.

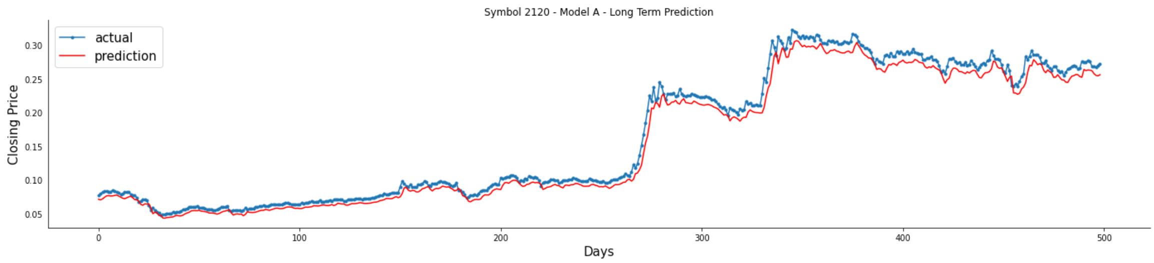


Figure 6.110: Model A - Symbol 2120 - Actual Close versus Predicted Close.

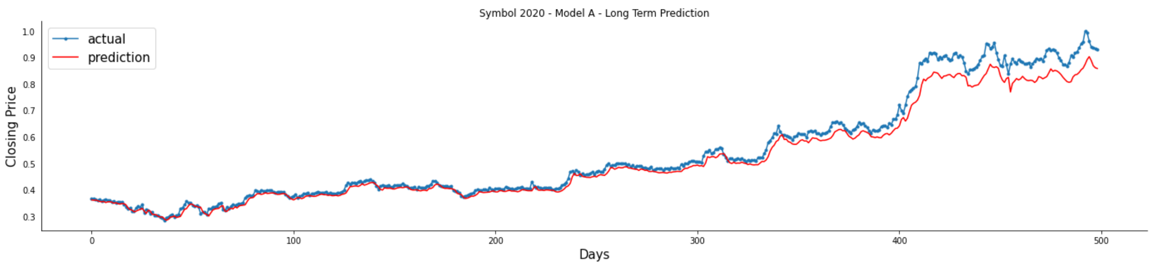


Figure 6.111: Model A - Symbol 2020 - Actual Close versus Predicted Close.

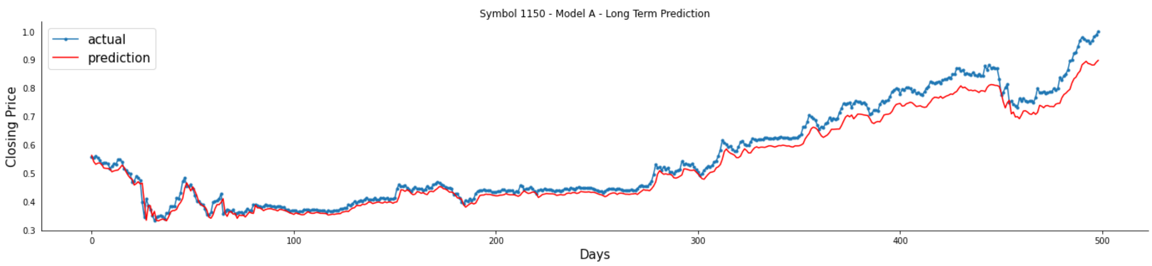


Figure 6.112: Model A - Symbol 1150 - Actual Close versus Predicted Close.

6.4.2.2 Model B Evaluations

Table 6.25 illustrates the Long Term Mean Absolute Error Percentage of predictions of the selected five stocks with Model B. Mean Absolute Error Percentages are ranging from 2% to 20% in the model. Symbol 4050 achieved the lowest error percentage of 2.51% from the model but no symbol achieved its best performances with Model B which feeds both OHLC and Technical Indicator data to the neural network.

Table 6.25: Model B – Long Term Stock Evaluations

Model	**Symbol Name**	**Mean Squared Error**	**Root Mean Squared Error**	**Mean Absolute Error**	**Mean Absolute Percentage Error**
LSTM	4050	0.00017285	0.01234363	0.00961747	2.51039338
GRU	4050	0.00029421	0.01518745	0.01022673	2.67582130
LSTM + GRU	4050	0.00037991	0.01479176	0.01244027	3.13596296
LSTM	4040	0.00006783	0.00823570	0.00833772	2.97837678
GRU	4040	0.00007528	0.00867659	0.00927284	3.26737784
LSTM + GRU	4040	0.00007173	0.00846923	0.00893738	3.76738923
LSTM	2120	0.00326493	0.05713954	0.05656584	21.05781364
GRU	2120	0.00079436	0.02818443	0.02727666	10.18725586
LSTM + GRU	2120	0.00067159	0.02593303	0.02443575	9.21879292

LSTM	2020	0.00085559	0.02928727	0.02521292	2.79865026
GRU	2020	0.00089124	0.02817450	0.02367180	2.61755753
LSTM + GRU	2020	0.00110611	0.03342376	0.02680853	3.02240062
LSTM	1150	0.00212198	0.04606499	0.03684894	5.29747293
GRU	1150	0.00250876	0.05008756	0.03774729	5.35839820
LSTM + GRU	1150	0.00206422	0.02577251	0.02923531	4.17684054

Figure 6.113 to Figure 6.117 visually illustrates the performance of Model B with each Stock Symbol. Figures have shown how Predicted Price from the Model B stacks up with Actual Price of the Stock. Symbol 4050 in Figure 6.113 show the best prediction out of the five symbols.

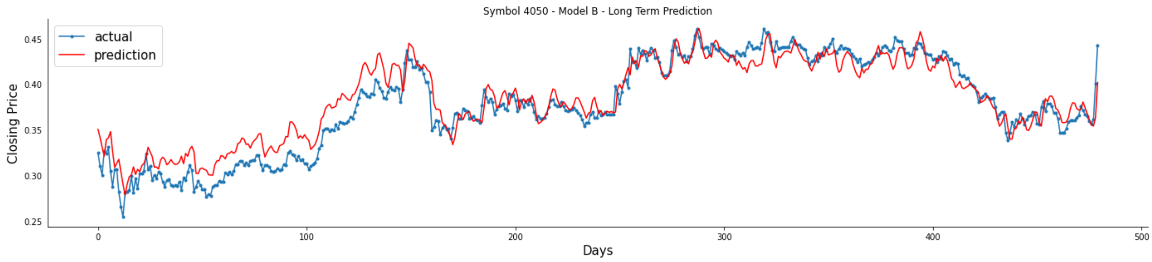


Figure 6.113: Model B - Symbol 4050 - Actual Close versus Predicted Close.

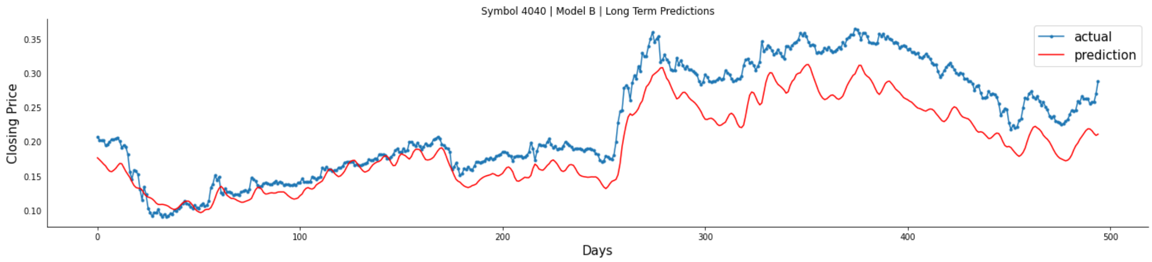


Figure 6.114: Model B - Symbol 4040 - Actual Close versus Predicted Close.

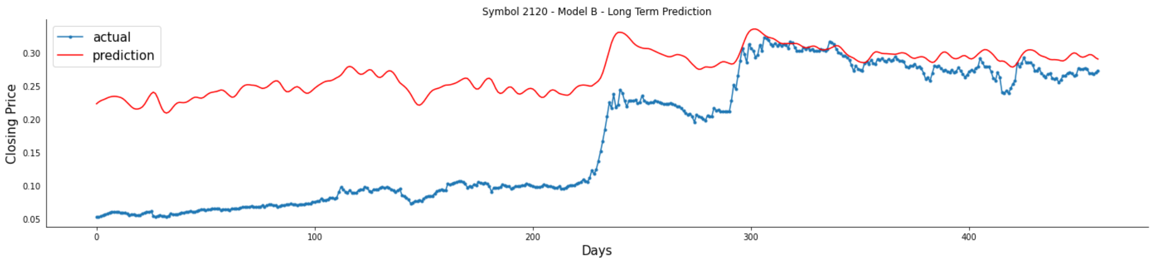


Figure 6.115: Model B - Symbol 2120 - Actual Close versus Predicted Close.

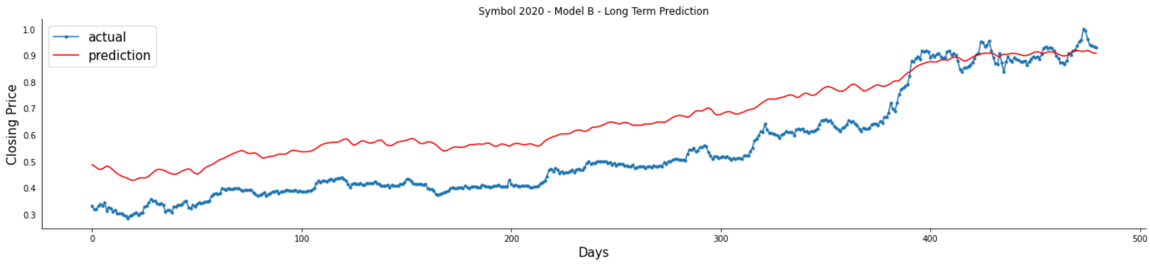


Figure 6.116: Model B - Symbol 2020 - Actual Close versus Predicted Close.

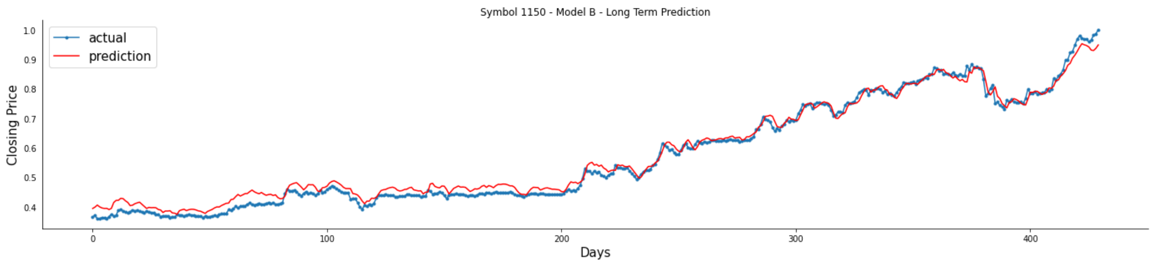


Figure 6.117: Model B - Symbol 1150 - Actual Close versus Predicted Close.

6.4.2.3 Model C Evaluations

Table 6.26 illustrates the Long Term Mean Absolute Error Percentage of predictions of the selected five stocks with Model C. Mean Absolute Error Percentages are ranging from 1 % to 12% in the model but no stock has registered its best prediction accuracies with Model C which feeds both OHLC and Technical Indicator data via a PCA Layer to the neural network. Model C generally achieved best accuracies when the network comprised of both LSTM and GRU units. Symbol 4040 achieved the best performance in Model C with a MAPE of 1.97%.

Table 6.26: Model C – Long Term Stock Evaluations

Model	Symbol Name	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	4050	0.00022130	0.01473000	0.01138787	2.92860627
GRU	4050	0.00045041	0.02099494	0.01674993	4.40052414
LSTM + GRU	4050	0.00021866	0.01478722	0.00951884	2.21338460
LSTM	4040	0.00006574	0.00810795	0.00783883	2.57838373
GRU	4040	0.00006739	0.00820907	0.00697838	2.67377392
LSTM + GRU	4040	0.00005189	0.00720366	0.00569374	1.97837737
LSTM	2120	0.00112946	0.03360744	0.02449503	12.74136353
GRU	2120	0.00023865	0.01550305	0.01369001	12.89622879
LSTM + GRU	2120	0.00010864	0.01044093	0.00823572	6.15046692

LSTM	2020	0.00253623	0.04828549	0.03564240	5.68879986
GRU	2020	0.02009409	0.13549294	0.09725504	14.92763901
LSTM + GRU	2020	0.00297228	0.05190579	0.03125145	4.45997953
LSTM	1150	0.00168212	0.03812021	0.02937290	5.09484911
GRU	1150	0.00106645	0.03265657	0.02791660	5.62400627
LSTM + GRU	1150	0.00119293	0.03195679	0.02421245	4.21348143

Figure 6.118 to Figure 6.122 visually illustrates the performance of Model C with each Stock Symbol. Figures have shown how Predicted Price from the Model C stacks up with Actual Price of the Stock. Predicted Price curve is very similar to the actual price curve indicating a good model performance over the 500 day period.

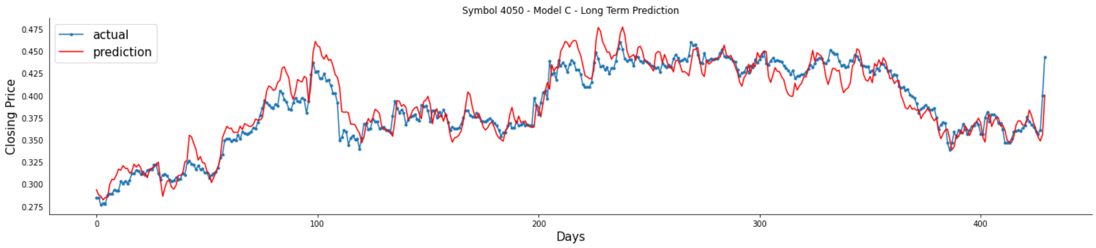


Figure 6.118: Model C - Symbol 4050 - Actual Close versus Predicted Close.

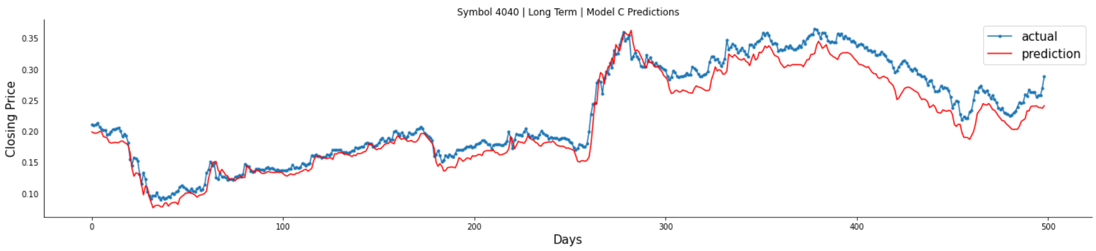


Figure 6.119: Model C - Symbol 4040 - Actual Close versus Predicted Close.

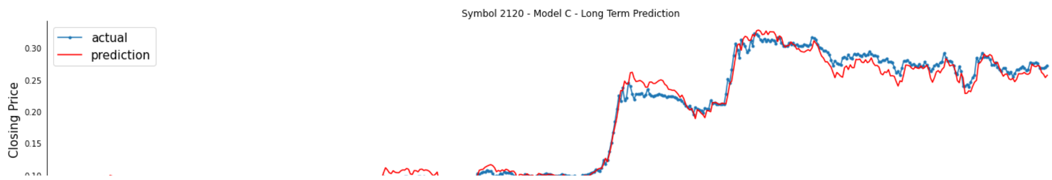


Figure 6.120: Model C - Symbol 2120 - Actual Close versus Predicted Close.

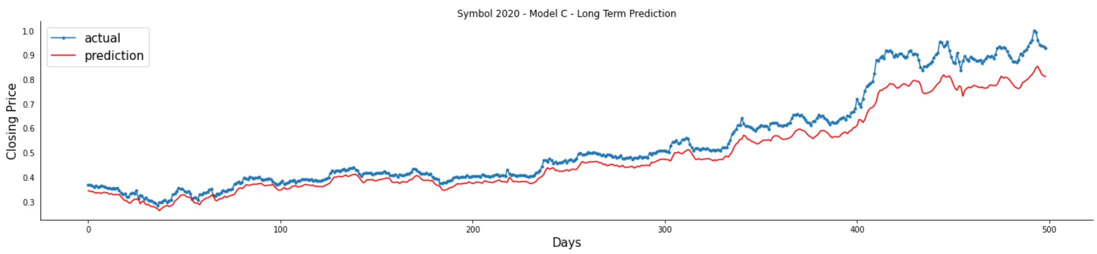


Figure 6.121: Model C - Symbol 2020 - Actual Close versus Predicted Close.

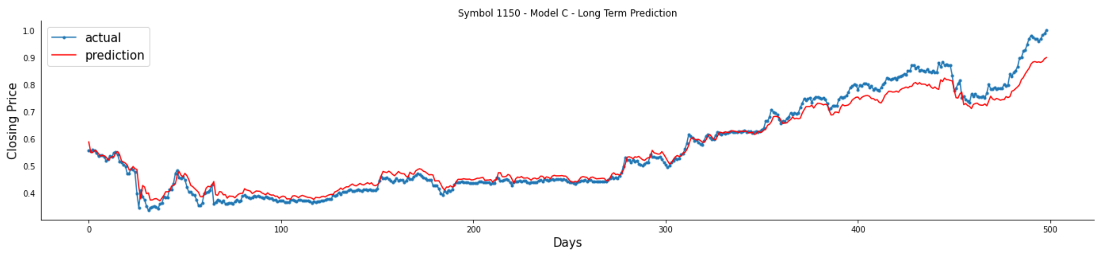


Figure 6.122: Model C - Symbol 1150 - Actual Close versus Predicted Close.

6.4.2.4 Model D Evaluations

Table 6.27 illustrates the Long Term Mean Absolute Error Percentage of predictions of the selected five stocks with Model D. Mean Absolute Error Percentages are ranging from 2 % to 7% in the model. Every Stock registered its best prediction accuracies with Model D .It is the model which feeds both OHLC data directly and Technical Indicator data via a PCA Layer to the neural network. Model D generally achieved best accuracies when the network comprised of both LSTM/GRU units and was the best model for Long term predictions in stocks. Symbol 4040 recorded the best MAPE of 1.87 % for the Model D and it was the best overall performance for any symbol in Long Term Predictions.

Table 6.27: Model D – Long Term Stock Evaluations

Model	Symbol Name	Mean Squared Error	Root Mean Squared Error	Mean Absolute Error	Mean Absolute Percentage Error
LSTM	4050	0.00031577	0.01780946	0.01352910	3.59236073
GRU	4050	0.00033494	0.01823057	0.01513462	2.96194625
LSTM + GRU	**4050**	**0.00022114**	**0.01483727**	**0.01189999**	**2.20515084**
LSTM	4040	0.00005684	0.00753911	0.00583927	2.53937372
GRU	4040	0.00005794	0.00761178	0.00537293	2.43539284
LSTM + GRU	**4040**	**0.00005264**	**0.00725519**	**0.00487832**	**1.87367237**
LSTM	2120	0.00023486	0.01503936	0.01088373	5.99226856

GRU	2120	0.00041855	0.02008797	0.01430561	7.31680441
LSTM + GRU	**2120**	**0.00011445**	**0.01065139**	**0.00779132**	**5.27735996**
LSTM	2020	0.00119087	0.03450895	0.02789235	5.06854534
GRU	2020	0.00124466	0.03527978	0.02922664	5.45766687
LSTM + GRU	**2020**	**0.00026200**	**0.01586193**	**0.01213392**	**2.22762346**
LSTM	1150	0.00052793	0.02261971	0.01919956	3.96830130
GRU	1150	0.00092826	0.03046733	0.02349670	4.16910219
LSTM + GRU	**1150**	**0.00116068**	**0.03142811**	**0.02325195**	**3.91113544**

Figure 6.123 to Figure 6.127 visually illustrates the performance of Model D with each Stock Symbol. Figures have shown how Predicted Price from the Model D stacks up with Actual Price of the Stock. Predicted Price is very similar to the actual price curve indicating a good model performance over the 500 day period.

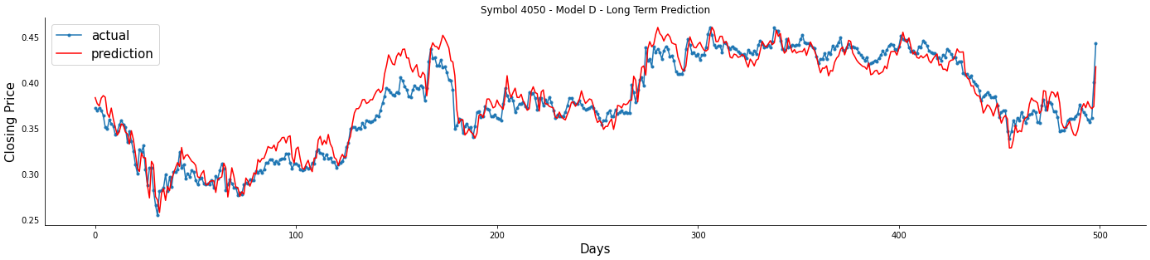


Figure 6.123: Model D - Symbol 4050 - Actual Close versus Predicted Close.

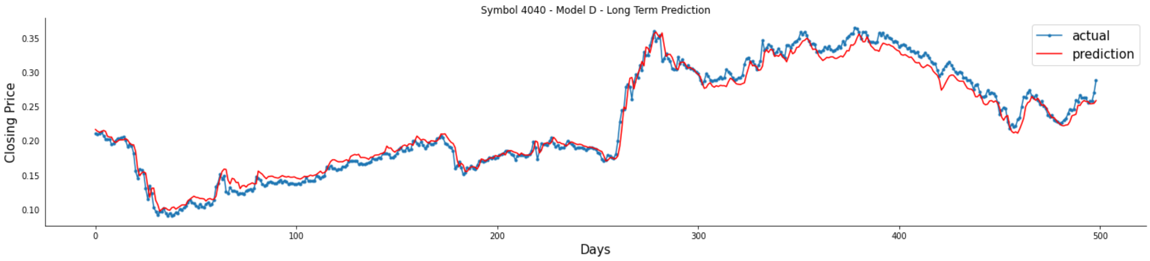


Figure 6.124: Model D - Symbol 4040 - Actual Close versus Predicted Close.

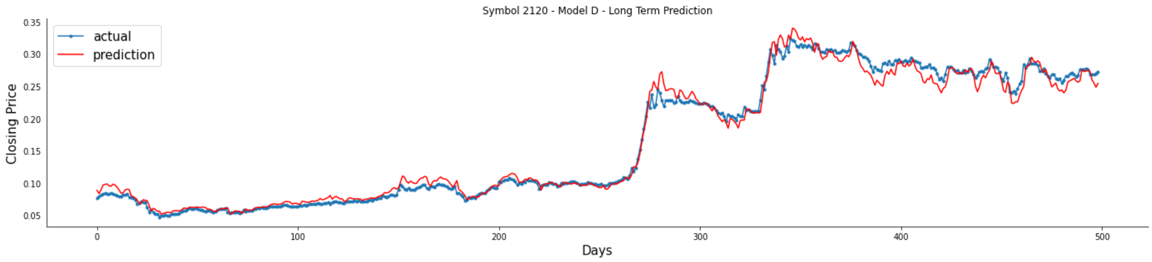


Figure 6.125: Model D - Symbol 2120 - Actual Close versus Predicted Close.

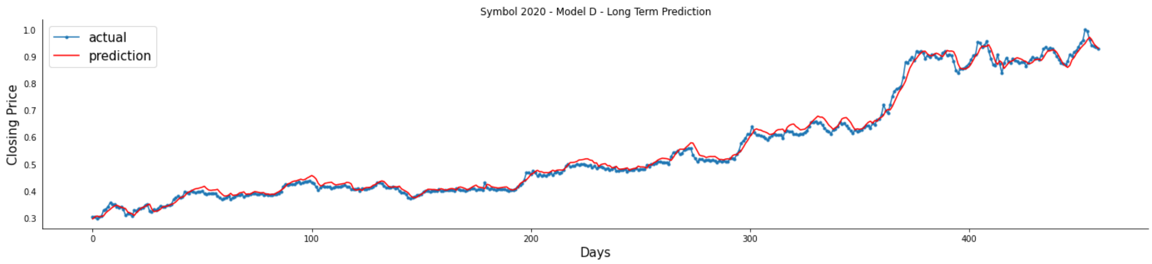


Figure 6.126: Model D - Symbol 2020 - Actual Close versus Predicted Close.

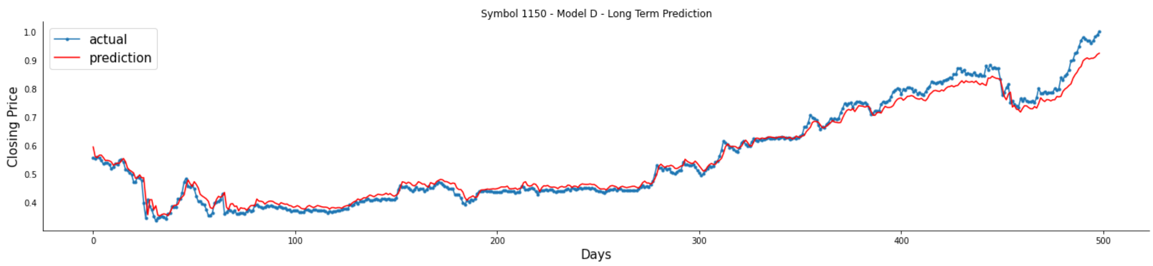


Figure 6.127: Model D - Symbol 1150 - Actual Close versus Predicted Close.

7. CONCLUSION AND RECOMMENDATION

Index data of Tadawul All Share Index and five fast moving stocks in Tadawul Stock Exchange was used in the research. A research to predict stock and index prices using deep learning techniques have never been performed at Tadawul Stock Exchange which is the main stock exchange in Middle East.

Research used below timelines for the prediction purposes.

1. Short Term Prediction – 25 days into the future

2. Medium Term Prediction – 100 days into the future.

3. Long Term Prediction – 500 days into the future.

Research clearly showed that hybrid network of using both LSTM and GRU in predicting the closing prices of both Indices and Stocks recorded best performances. TASI Index recorded a best MAPE of 0.94% in Short Term , 1.15% in Medium Term and 1.53% in Long Term using the model which feeds OHLC data directly and Technical Indicator data directly to the neural network (Model D). The best performances were obtained only when the model used hybrid network of LSTM and GRU units.

Every stock except stock 2120 in short term recorded individual best performances in terms of MAPE using the model which feeds OHLC data directly and Technical Indicator data indirectly to the neural network (Model D). All of the best performances were obtained when model used a hybrid network comprising of both LSTM and GRU units. All Stocks achieved best performances close to 1% in short term, 1 to 3 % in medium term and 1 to 5% in long term.

Short Term Predictions of TASI Index using the model which feeds OHLC data directly and Technical Indictor data via PCA Layer (Model D) shows less than 1% Mean Absolute Percentage Error. It is considered as a very reliable benchmark. The best performances were achieved when the model used hybrid network of GRU and LSTM layers. Stocks does not show higher accuracy percentages as Index and their error percentage ranges 1% to 3%. Lowest Error rates for Stock Prediction are shown with the model which fed OHLC data directly and Technical Indicator data via a PCA Layer (Model D) and with the model which feeds OHLC data directly to the neural network (Model A). Hybrid Networks showing lowest MAPE for 4 out of 5 stocks and only one stock recorded the lowest MAPE with LSTM only model.

TASI index Medium Term Prediction recorded a MAPE ranging from 0.8 % to 3 % which gave a good accurate prediction of the Mid Term Index Price movements. Five Stocks indicated a MAPE ranging from 1% to 3% which is significantly closer to Index MAPE values. Model which fed OHLC data directly and Technical Indicator data via a PCA Layer (Model D) achieved the best results when it is used with hybrid network.

TASI index Long Term Prediction shows MAPE ranging from 1% to 4% which showed significant capabilities of the model to predict more accurate values. Best Performances was given by the model which fed OHLC data directly and Technical Indicator data via a PCA Layer (Model D) when it is used with Hybrid LSTM and GRU Layers. All Stocks recorded MAPE from 1% to 5% which is slightly worse than the TASI Index Error Percentage Values. But still they were high accurate predictions in long term. Three out of Five Stocks showed MAPE of less than 2% almost equaling the TASI Index Prediction Percentages.

All Models were trained with quantitative data of an Index or a Stock. The Closing Price movements are not solely effected by market demand but also from external socio-economic factors as well. Variations due to these external factors were not taken into account by the model. Index Price indicates the cumulative quantitative performance of group of stocks. Any drastic effect on an Individual Stock's Close Price due to socio-economic reasons has been weigh down by the cumulative addition of the other stocks which contribute to the calculation of Index Price. All the model types discussed in the research managed to predict Index Prices with less error rate than Stock prices due to this reason. A Model cannot model the sudden changes in a Stock's Close Price due to their related company performance. Hence the error rate is high when compared with Indices.

Short Term Predictions registered relative high error rates as all four models fail to grab short term fluctuations due to external socio-economic factors. Socio-Economic changes effect the short term fluctuations more than that of medium or long term fluctuations. Medium and Long Term Predictions were much less affected by these external factors and dominated by normal market demand and supply. All four models performed better in predicting medium and long terms due to this reason.

Model which fed OHLC data directly and Technical Indicator data via PCA Layer (Model D) showed the most accurate results. The PCA Layer helped to reduce the dimensionality of Technical Indicators and removed more correlated features. Then model complexity was dropped down resulting in better performances than other models.

Finally, the research firmly shows the effective usage of LSTM and GRU Hybrid Neural Network Models for Stock and Index Price Prediction using OHLC and Technical Indicator data. It also shows very accurate predictions for TASI Index and Stocks in Tadawul Stock Exchange which has been never been a part of a similar study.

REFERENCES

[1] "Stock Market | Investopedia", Investopedia. [Online]. Available: https://www.investopedia.com/terms/s/stockmarket.asp. [Accessed: 05- Jun- 2021]

[2]"Market Index", Investopedia. [Online]. Available: https://www.investopedia.com/terms/m/marketindex.asp. [Accessed: 05- May- 2021]

[3] Guo Z, Wang H, Liu Q, Yang J. A Feature Fusion Based Forecasting Model for Financial Time Series. Plos One. 2014; 9(6):172±200.

[4] V. Cherkassky, "The Nature Of Statistical Learning Theory~", IEEE Transactions on Neural Networks, vol. 8, no. 6, pp. 1564-1564, 1997.

[5] Refenes AN, Zapranis A, Francis G. Stock performance modeling using neural networks: A comparative study with regression models. Neural Networks. 1994; 7(2):375±88.

[6] Yoon Y, Margavio TM. A Comparison of Discriminant Analysis versus Artificial Neural Networks. Journal of the Operational Research Society. 1993; 44(1):51±60.

[7] Hinton GE, Salakhutdinov RR. Reducing the Dimensionality of Data with Neural Networks. Science.

2006; 313(5786):504±7. https://doi.org/10.1126/science.1127647 PMID: 16873662

[8] Bengio Y, Courville A, Vincent P. Representation Learning: A Review and New Perspectives. IEEE Transactions on Pattern Analysis & Machine Intelligence. 2013; 35(8):1798±828.

[9]Cavalcante RC, Brasileiro RC, Souza VLF, Nobrega JP, Oliveira ALI. Computational Intelligence and Financial Markets: A Survey and Future Directions. Expert Systems with Applications. 2016; 55:194±211

[10] G. Hinton, "Deep belief networks", Scholarpedia, vol. 4, no. 5, p. 5947, 2009.

[11] Alexiei Dingli and Karl Sant Fournier, "Financial Time Series Forecasting – A Deep Learning Approach," International Journal of Machine Learning and Computing vol. 7, no. 5, pp. 118-122, 201

[12]Bao W, Yue J, Rao Y (2017) A deep learning framework for financial time series using stacked autoencoders and long-short term memory. PLoS ONE 12(7): e0180944. https://doi.org/10.1371/journal.pone.0180944

[13]Chu KL, Sahari KSM. Behavior recognition for humanoid robots using long short-term memory. 2016;13(6):172988141666336.

[14]Hinton G, Deng L, Yu D, Dahl GE, Mohamed A, Jaitly N, et al. Deep Neural Networks for Acoustic Modeling in Speech Recognition. IEEE Signal Processing Magazine. 2012; 29(6):82±97

[15] Palangi H, Ward R, Deng L. Distributed Compressive Sensing: A Deep Learning Approach. IEEE Transactions on Signal Processing. 2016; 64(17):4504±18.

[16] X. Li, L. Yang, F. Xue and H. Zhou, "Time series prediction of stock price using deep belief networks with intrinsic plasticity," 2017 29th Chinese Control And Decision Conference (CCDC), Chongqing, China, 2017, pp. 1237-1242, doi: 10.1109/CCDC.2017.7978707.

[17] Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling, Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, Yoshua Bengio.

[18]"Investopedia", Investopedia, 2022. [Online]. Available: https://www.investopedia.com. [Accessed: 05- Apr- 2022]

[19] J. Chung, C. Gulcehre, K. Cho and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling", arXiv.org, 2022. [Online]. Available: https://arxiv.org/abs/1412.3555. [Accessed: 05- Apr- 2022]