

Prediction of Stock Market using Stochastic Neural Networks

Bhanu Teja Reddy, Usha J.C

Abstract— The primary objective of investors and stockbrokers is to make profits by being able to predict the financial markets. However, forecasting is a complex task since the financial markets have a complicated pattern. This study addresses the direction of the stock price index for Japanese Nikkei 225. The research compares two prediction models, i.e., the Stochastic Neural Networks (SNN) and fusion of Long-Short Term Memory and Stochastic Neural Networks (LSTM - SNN) for predicting the index. The input layer includes computation of fifteen technical indicators using stock market parameters (open, high, low, close prices, and volume). Accuracy of each of the prediction models was evaluated using price and trend performance metrics. The evaluation was carried out for historical data from 23rd January 2007 to 30th December 2013 of the Tokyo Stock Exchange (TSE). The experimental outcomes recommend that for the SNN, the model gave an accuracy of 85.37% and hybrid of LSTM – SNN gave accuracy of 86.28%. The increase in the accuracy of LSTM – SNN was due to the introduction of LSTM layer. Experimental outcomes also illustrate that the performance of both the prediction models progress when these technical indicators are added to the input layer of the proposed models.

Index Terms— Artificial Neural Networks, Long Short-Term Memory (LSTM), Stochastic Neural Networks (SNN), Stock Market Prediction, Tokyo Stock Exchange (TSE)

I. INTRODUCTION

A. Motivation

Efficient Market Hypothesis (EMH), one of the prominent theories in financial investments states that the prices of the assets entirely reflect the information which is already available, and it is not possible to defeat the market

Manuscript received September 01, 2019

Bhanu Teja Reddy, Financial Management, M.S Ramaiah University of Applied Sciences, Bengaluru, India, Mobile 9738630786, (e-mail: bhanu7teja@gmail.com)

Usha J.C, Financial Management, M.S Ramaiah University of Applied Sciences, Bengaluru, India, Mobile 9483219761

consistently [1]. The EMH theory is highly conflicted and often disputed. The perfect example would be shareholders like Warren Buffet has earned an immense amount of profits for a brief period by consistently defeating the market. Even though predicting the movement of stock price by non-automated methods and interpreting the chaotic market data may be a tedious task, with the approach of ML, ANN, DL, and GA techniques, big data and rise in computational capabilities and automated methods for predicting the prices of the stock are becoming achievable. ML, ANN, DL, and GA techniques are capable of training a function by reading the data without explicitly being computed [2]. However, the time-series of the stock is not a function that can map easily. It tends to best depict as a random walk, where it makes the prediction and feature engineering much harder [3].

B. Introduction

Predicting the growth of financial instruments such as stocks has evident economic advantages, and abundant approaches are in place trying to accomplish this [4]. Lately, Machine Learning (ML) approaches have been used to model financial stock prices. These models are evaluated and compared for different financial stocks using different performance metrics to know the efficiency of these Machine learning models and have accomplished mixed success [5]. A comparative assessment of these approaches is timely, given the scope of artificial neural networks for the prediction of stock markets [6].

Fortune Hunters and Academics have been trying to forecast the financial market [7]. It is exciting and beneficial to predict financial markets and individual stocks, as one can obtain both financial advantages and economic understanding, driven by several variables. Stocks are interesting to predict [8]. Investigators are not able to decide on the certainty of the prediction of financial markets. Studying and understanding the unpredictable market is exciting as well as a challenging task [9].

Techniques like Deep Learning (DL), a machine learning subdivision, training can be started using the raw data and features will be created automatically when learning the neural network [7]. Deep learning techniques are among the popular methods used to identify stock trends from large amounts of data, but there is so far, no such algorithm or model that could consistently predict stock prices [6]. The objective of the stock market in like manner is unusual, which may rely upon the long- and short-term future state. The state is terrible and furthermore unavoidable for the

examiner when the stock market held as a speculation device [10]. The fundamental point is to reduce this unusualness, and the stock market anticipation is held in this process [11].

The remainder of this manuscript is structured as follows, Section I comprises of introduction on stock market prediction, Section II contains the related work for stock prediction, Section III contains the methods and methodology of the study, Section IV consists of the results and discussions, section V consists of conclusion of the research work with future directions.

II. RELATED WORKS

A. Literature Review

Hagenau, Liebmann and Neumann, 2013 studied on textual data in financial news for predicting the stocks. In accord with prior text mining models, several feature selection methods were employed for extracting the relevant features. The system aimed for achieving better classification efficiencies, yet, the system failed on semantically related features and thus decreased the over-fitting problem in machine learning approaches [12].

Evans, Pappas & Xhafa, 2013 studied the FOREX transactions models for prediction and decision-making systems. These kinds of data explored various irrelevant noises on acquired data. The produced forecasting models often dealt with feed-forward neural networks with the backpropagation architecture. Genetic algorithms were also used for optimal network topology. Most of the topologies yielded better network approximation and generalization [13].

Li et al., 2014 studied the platform of trading signal mining which influences the information sources, namely, news articles about the market, and the process of the stock tick. In order to determine the better prediction accuracy and prediction speed with core algorithms like B-ELM and K-ELM were explored. Li et al., 2014 studied on trading signal mining platform that influences the information sources, namely, market news articles, and the stock tick process. For determining the better prediction accuracy and prediction speed with core algorithms like B-ELM and K-ELM were explored [14].

Two-stage fusion models such as Support Vector Regression (SVR), Random Forest (RF), ANN were explored by Patel et al., 2015a. The analysis showed that hybrid models of two stages performed better in comparison with prediction models having a single stage. It is crucial in practical cases of RF and ANN than SVR. Most of the SVR models helped to achieve better prediction rate. The authors proved that the proposal of two-stage prediction yielded better information sources [15].

Patel et al., 2015b presented the difficulty of predicting the stock price with Indian stocks. The forecasting techniques like Naive Bayes, ANN, SVM, and RF were studied for predicting the stock trading data. The prediction of two input methods using past data from 2003 to 2012 on reliance industries and the Infosys Ltd. The authors also dealt with CNX nifty and S& P Bombay stock exchanges [16].

Rather, Agarwal and Sastry, 2015 discussed the robust and novel hybrid model using stock return prediction. It composes of two linear models, namely, the exponential smoothing model and autoregressive model, namely recurrent neural networks. Recurrent neural networks provide better prediction models that resolved prediction issues [17].

Wang and Wang, 2015 studied the stochastic time analysis of active neural networks using principal component analysis analyzed the indexes of DJIA, S&P500, HS300, and SSE. The models such as PCA-BPNN, STNN, and BPNN were evaluated. Empirical studies of defining the time series demonstrated that forecasting the precision values defines the actual prices of the stock market [18].

Qiu, Song, and Akagi, 2016 explored the Nikkei 225 index utilizing ANN. Here, 71 variables were collected on the Japanese stock market and analyzed using fuzzy surfaces. Likewise, monthly data analysis using 18 variables were collected to predict the stock market returns [6].

Moghaddam & Esfandyari, 2016 discussed the NASDAQ exchange rate of variant parameters. The data obtained from the rate of exchange for the final working days were analyzed using OSS training method and TANGSIG of the network topology comprising of hidden layers with 20-40-20 neurons of R2 values of 0.9408 to validate the dataset. The number of neurons 40-40 was studied for OSS training methods [19].

The research reported in Cocianu and Grigoryan, 2016 analyzed the stock market prediction issues. Variable selection models using neural networks and SVM that optimized the effectiveness of the variables obtained using stock market indicators. Error minimization of the stock price deals with error indicators of a historical selection of the data [20].

Di Persio and Honchar, 2016 explored the ANN combined with MLP, CNN, and RNN used for forecasting the stock market price movements. The trained results are compared with the S&P 500 index of the CNN and variant time-series predictions. Likewise, Wavelet + CNN algorithm was explored under neural network approaches. Feature pre-processing models determined as essential parts in stock price forecasting [21].

Zhong and Enke (2017) studied the effective everyday direction of stock market return utilizing data cleaning and preprocessing techniques and explored better simulation results [22]. Krauss, Do and Huck, 2017 explored the statistical approach using deep neural networks, gradient boosted trees, and random forests, which analyzed over S&P 500 data constituents [23]. Chong et al. (2017) described the systematic evaluation of deep learning networks for stock market assessment and prediction. Prior knowledge about a comprehensive set of raw data is being explored using DLNs. Most of the DLNs alter the activation function, the structure of the network, activation function with better data representation [9].

Mohapatra, Majhi, and Satapathy, 2017 showed earning based on incrementing and diffusion to explain FLANN predictor. The system reduced the computational burden of neural networks from the hidden layer, along with the propagation of error. It also decreased the computing cost and better rate of error convergence at variant future values

prediction. Singh and Srivastava, 2017 worked using deep learning for stock data forecasting. They demonstrated PCA with deep learning using google datasets that proved a better accuracy rate compared to conventional neural networks with PCA [24].

Fischer and Krauss, 2018 discussed an extended model for short term memory and suggested for financial markets of large-scale from the S&P 500, between December 1992 and October 2015. This study explored their contributions in three ways; namely, large-scale empirical applications, sensible features in the form of 240 days return sequences and then pre-processing for training model. It also helps to deduce a prediction based trading strategy [25].

Song, Zhou, and Han, 2018 explored the significance of dynamically developed algorithm that evaluated the volatile stock market. Based on the share market price, the investor selects the buy or sell a stock with extremely conditioned parameters. This study practices a model-independent approach to show the hidden dynamics of stock market data using various deep learning techniques such as RNN, LSTM, and GRU [26]. Zhang et al., 2019 explored the price trend forecasting models using hidden Markov models. It solved parameters estimation and decoding of higher-order HMM [27].

B. Research Gap

- The combination of technical indicators like On-Balance-Volume(OBV), Bollinger Bands (BB), Rate of Change (ROC) and Disparity (DIS) was not considered in previous studies for stock market prediction [4] [15][16].
- Previous studies have considered two different models for predicting the price and trend of the stocks [15][16].
- Scant research was done on Stochastic Neural Networks and fusion of Long Short-Term Memory and Stochastic Neural Networks.

III. METHODS AND METHODOLOGY

A. Stochastic Neural Networks (SNN)

SNN is a class of deep learning approaches constructed through the introduction of random variants in the network, either by providing stochastic transfer functions to the neurons of the network or by offering them stochastic weights. This makes them helpful tools for issues of optimization, as random fluctuations assist it in escaping local minima [18]. This Model is built using python Tensorflow library. The weights are set using NumPy initially. Adam optimiser is used to optimise the data. The Boltzmann machine is an example of SNN that uses stochastic transfer features. Each neuron is rated as binary, and the likelihood of firing it depends on the network's other neurons. SNN is used for risk management, bioinformatics oncology, and other comparable areas. The SNN has found its application in financial market prediction due to their offering of stochastic weights.

B. Long Short-Term Memory - Stochastic Neural Networks (LSTM – SNN)

LSTM is a class of neural network whose output does not only depend on present input but also the previous inputs. This Model is built using python Tensorflow library. LSTM networks are well-suited to classifying, processing, and making predictions based on time series data, which is a critical property considering that stock prices are time series. SNN is built by introducing random variations to the network. LSTM-SNN is hybridised to achieve better output by gaining the advantage of the memory function of LSTM and randomness of SNN. However, that is not possible if the network gets stuck in local minima. This neural network has one input layer one output layer, and two hidden layers, among which one of them is an LSTM layer and another one, is an SNN layer. The weights are set using NumPy initially. Adam optimiser is used to optimise the data. The LSTM – SNN applied for tasks such as oncology, connected handwriting recognition, and bioinformatics, robot control.

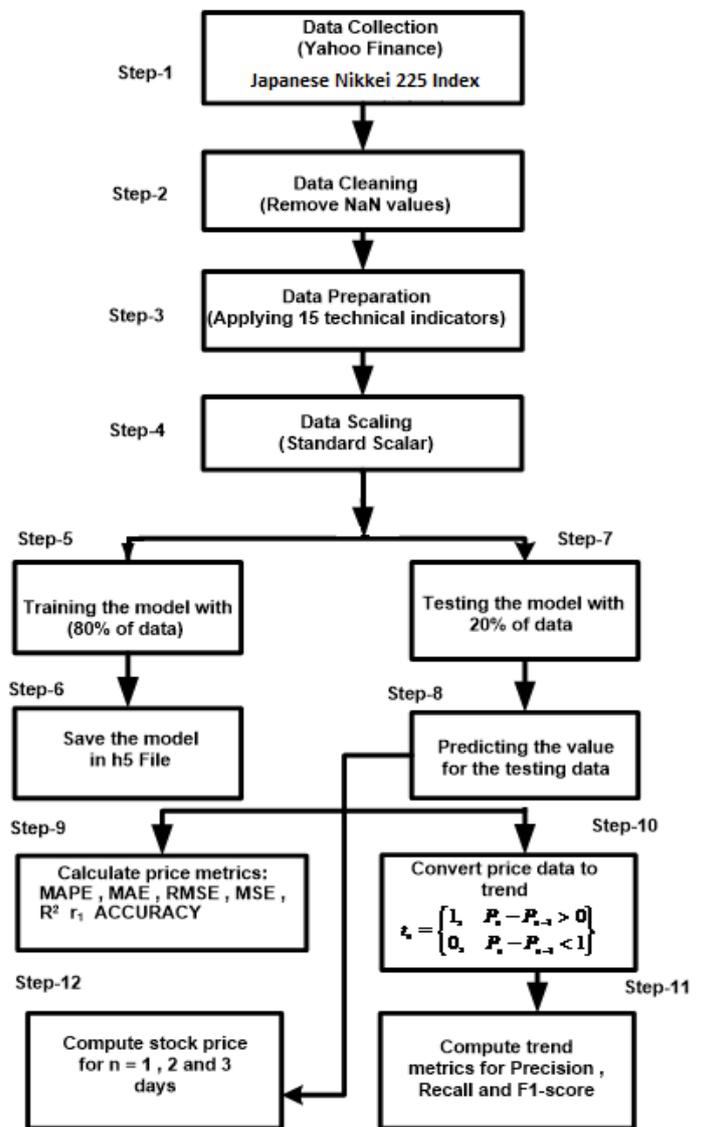


Fig. 1: Flowchart of SNN and LSTM – SNN methodology

C. Methodology

Step 1_Data Collection: Historical stock prices of the Japanese Nikkei 225 Index is considered to challenge the study of Qiu and Song, 2016 [6]. The data is downloaded from yahoo finance for the period January 23rd 2007 to December 30th 2013 the data is considered based on Qiu and Song, 2016 for their research.

Step 2_Data Cleaning: The data cleaning is done using python function called dropna to remove all the null values called NaN (Not-a-Number) from the downloaded historical data for normalizing the data, and the file was saved as Data.xls

Step 3_Data Preparation: Data preparation is done using python by developing an algorithm for fifteen technical indicators (as shown in Figure 2) and expressions mentioned in Table 1 for preparing data for the input layer of the models. The data preparation is done on Data.xls file by using stock parameters (open,high,low,close, volume) and is saved as Input.xls.

Step 4_Data Scaling: The primary importance behind the inclusion of the conventional scalar is that when considering a regression issue such as stock price forecast with non-linear time series, it helps to effectively scale information. The standard scalar (Stdscale) standardizes the data points according to the mean value 0 and standard deviation 1 during the reshaping phase. The standard scalar function was applied in python using Data.xls file and was saved as ScaledData.xls

Step 5_Training: The data is split into 80% and 20%, where 80% of the data was used for training. Where the model tries to train itself from the data applied in the input layer to recognise the pattern of the stock market for future predictions, here two hidden layers HL1 and HL2 each of size 500. Batch size of 30 and epoch size of 200 for predictions.

Step 6_h5 file: The data trained is saved in h5 file for easy recognising of pattern whenever required for prediction.

Step7 and 8_Testing and Evaluating: The data trained is tested for prediction of the stock market and evaluated the model’s performance metrics for both trend and price.

Step 9_Price metrics: Price metrics are evaluated to know the performance of the model for price prediction of the index. The price metrics considered are MAPE, MAE, MSE, RMSE, R², r₁ for evaluation.

Step 10_Conversion: Price data is converted to trend data for trend prediction. The formula used for conversion from price data to a trend is shown below

$$T_n = 1, P_n - P_{n-1} > 0$$

$$0, P_n - P_{n-1} < 0$$

Where T_n is Today’s Price
 P_n is previous day price
 P_{n-1} is before previous day price

Step 11_Trend metrics: Trend metrics are evaluated to know the performance of the model for trend prediction of the index. The trend metrics considered are Precision_{positive}, Precision_{negative}, Recall_{positive}, Recall_{negative}, Accuracy, and F₁ score for evaluation.

Step 12_Prediction: Nikkei 225 index is predicted for the three consecutive days by using the formula shown below

$$T1 = dataset[dataset.length] - P1 > 0$$

$$T2 = P1 - P2 > 0$$

$$T3 = P2 - P3 > 0$$

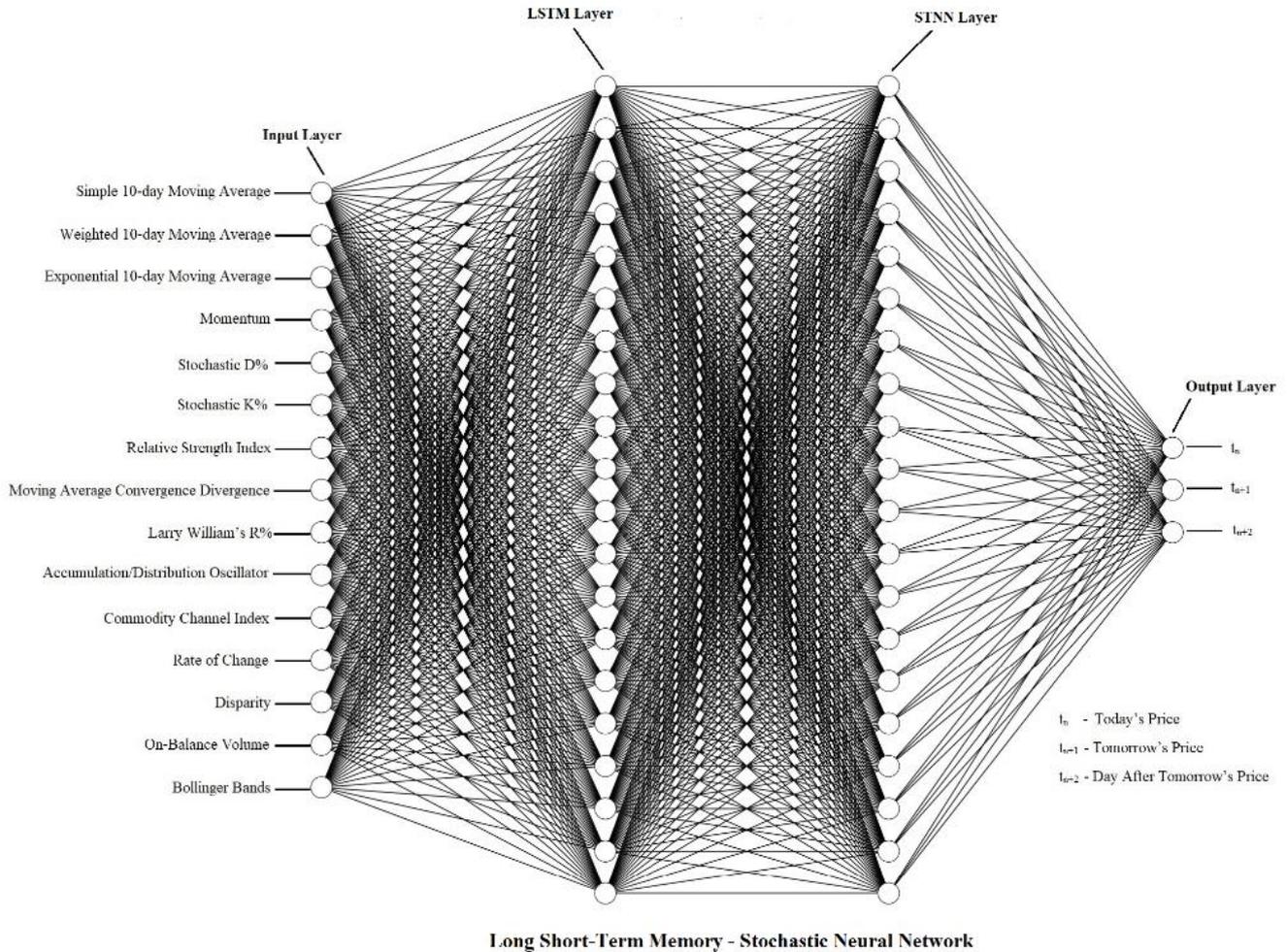


Fig. 2: Long Short Term Memory – Stochastic Neural Network (LSTM -SNN) model diagram

The successive section further emphasizes the SNN flow. To explain the regression problem for the purpose of strengthening the stock prediction accuracy, which is mathematically shown with the following algorithm and also further illustrated with a flowing design.

D. Algorithm for LSTM – SNN

Start

[1] Procedure AllPhases()

Phase 0:

[2] Dataset = read
 (Date,Open,Low,High,Close,Adjusted,Close,Volume)

Phase 1:

[3] Dataset = calculate TI
 (SMA,WMA,EMA,MOM,STCK%,STCD%,RSI,MACD,
 WILLR%,A/D OSS,CCI,ROC,DISP,OBV,BB)
 [4] Dataset = window()
 [5] Model = prepare_model()

Phase LSTM - STNN:

[6] Input = layer(size = Dataset.shape[0])
 [7] L₁ = LSTM Layer(size=500,connect=Input)
 [8] L₂ = SNN Layer(size=500,connect=L₁)
 [9] Output = layer(size=3,connect=L₂)

[10] Batch_size = 30

[11] Epochs = 200

Phase 2:

[12] Dataset_train = dataset[start:0.8*dataset.length]

[13] Dataset_test = dataset[0.8*dataset.length:end]

Phase 3:

[14] Model = model.train(dataset_train)

[15] Predicted = model.predict(dataset_test)

[16] Actual = dataset[0.8*dataset.length][“close”]

Phase 4:

[17] For P_n in Predicted for n = 1,2,3

[18] If P_n – P_{n-1} > 0 goto 19 else goto 20

[19] T_n = 1 goto 21

[20] T_n = 0

[21] Repeat 20 till n = 3

[22] For A_n in Actual

[23] If A_n-A_{n-1} > 0 goto 24 else goto 25

[24] Actual_trend = 1 goto 26

[25] Actual_trend = 0

[26] Repeat 25

Phase 5:

[27] Calculate_metrix(MAPE,RMSE,MSE,MAE,r₁,R₂)

[28] Precision = calculate_precision(actual_trend,trend)

[29] F₁Score = calculate_F₁Score(actual_trend,trend)

[30] Recall = calculate_recall(actual_trend,trend)

[31] Accuracy = mean(Trend = Actual_trend)

Phase 6:

[32] P₁, P₂, P₃ = evaluateResults()

[33] T₁ = dataset[dataset.length] - P₁ > 0

[34] T₂ = P₁ - P₂ > 0

[35] T₃ = P₂ - P₃ > 0

end

The above highlighted computational and mathematical algorithm steps exhibit how the execution flow of the Siamese model will handle the regression problem, which is associated with non-linear time series.

E. Mathematical expressions for technical indicators

| Name of the TI | Mathematical Expressions |
|--|---|
| Simple Moving Average (SMA) | $\frac{C_t + C_{t-1} + \dots + C_{t-9}}{n}$ |
| Weighted Moving Average (WMA) | $\frac{(10)C_t + (9)C_{t-1} + \dots + (1)C_{t-9}}{n + (n-1) + \dots + (n-9)}$ |
| Exponential Moving Average (EMA) | $C_t - EMA_{t-1} \times \left(\frac{2}{n+1} + EMA_{t-1} \right)$ |
| Momentum (MOM) | $C_t - C_{t-9}$ |
| Stochastic K% (K%) | $\frac{C_t - LL_n}{HH_n - LL_n} \times 100$ |
| Stochastic D% (D%) | $\sum_{i=0}^{n-1} \frac{K_{t-i} \%}{10}$ |
| Relative Strength Index (RSI) | $100 - \frac{100}{1 + \left(\frac{\sum_{i=0}^{n-1} UP_{t-i} / n}{\sum_{i=0}^{n-1} DW_{t-i} / n} \right)}$ |
| Moving Average Convergence Divergence (MACD) | $MACD(n)_{t-1} + \frac{2}{n+1} \times (DIFF_t - MACD(n)_{t-1})$ |
| Larry William's R% (R%) | $\frac{H_n - C_t}{H_n - L_n} \times -100$ |
| Accumulation/ Distribution Oscillator (A/D) | $\frac{H_t - C_{t-1}}{H_t - L_t}$ |
| Commodity Channel Index (CCI) | $\frac{M_t - SM_t}{0.015D_t}$ |
| Rate of Change (ROC) | $\frac{C_t - C_{t-n}}{C_{t-n}} \times 100$ |
| On-Balance-Volume (OBV) | $OBV_{t-1} + \begin{cases} V_t & \text{if } C_t > C_{t-1} \\ 0 & \text{if } C_t = C_{t-1} \\ -V_t & \text{if } C_t < C_{t-1} \end{cases}$ |
| Disparity (DIS) | $\frac{C_t - SMA_n}{SMA_n} \times 100$ |
| Bollinger Bands (BB) | $SM_t \pm m \times \sigma[M_t]$ |

L_t is the low price; H_t is the high price, C_t is the close price, and V_t is the volume at time t, k is the time period of k-day (K%) exponential moving average, HH_n and LL_n and

implies highest high and lowest low for n days respectively. DIFF = EMA₍₁₂₎ - EMA₍₂₆₎, DW_t is the downward price change while UP_t means upward price change at time t. $M_t = \frac{H_t + L_t + C_t}{3}$, $SM_t = \frac{(\sum_{i=1}^n M_{t-i+1})}{n}$, m is the number of standard deviations (typically 2) and $\sigma[M_t]$ is the standard deviation of M_t.

F. Performance Metrics for trend prediction

Accuracy and f-measure are used to evaluate the performance of proposed models. Computation of these evaluation measures requires estimating Precision and Recall, which are evaluated from True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN).

| Name of the Trend Metrics | Mathematical Expressions |
|-------------------------------|---|
| Precision _{positive} | $\frac{TP}{TP + FP}$ |
| Precision _{negative} | $\frac{TN}{TN + FN}$ |
| Recall _{positive} | $\frac{TP}{TP + FN}$ |
| Recall _{negative} | $\frac{TN}{TN + FP}$ |
| Accuracy | $\frac{TP + TN}{TP + FP + TN + FN}$ |
| F ₁ Score | $\frac{2 \times Precision \times Recall}{Precision + Recall}$ |

Precision is the weighted average of precision positive and negative, while Recall is the weighted average of recall positive and negative. Accuracy and F-measure are estimated, respectively.

G. Performance Metrics for price prediction

Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), relative Root Mean Squared Error (rRMSE) and Mean Squared Error (MSE), r₁ is the correlation between the actual value and predicted value, R² is the coefficient of determination used to evaluate the performance of these prediction models. Formulas of these evaluations, as shown below:

| Name of the Price Metrics | Mathematical Expressions |
|--|--|
| Mean Absolute Percentage Error (MAPE) | $\frac{1}{n} \sum_{t=1}^n \frac{ A_t - F_t }{A_t} \times 100$ |
| Mean Absolute Error (MAE) | $\frac{1}{n} \sum_{t=1}^n A_t - F_t $ |
| Mean Square Error (MSE) | $\frac{1}{n} \sum_{t=1}^n (A_t - F_t)^2$ |
| Root Mean Square Error (RMSE) | $\sqrt{\frac{1}{n} \sum_{t=1}^n (A_t - F_t)^2}$ |
| Correlation Co-efficient (r ₁) | $\frac{\sum_{t=1}^n (A_t - \bar{A}) \times (F_t - \bar{F})}{\sqrt{\sum_{t=1}^n (A_t - \bar{A})^2 \times (F_t - \bar{F})^2}}$ |

| | |
|--|---|
| Coefficient of Determination (R^2) | $1 - \frac{\sum_{t=1}^n (A_t - F_t)^2}{TSS \sum_{t=1}^n (A_t - \bar{A})^2}$ |
|--|---|

Where A_t is the actual value and F_t is the predicted value and \bar{A} is the mean of A_t values and \bar{F} is the mean of F_t values.

IV. RESULTS AND DISCUSSIONS

A. Results

- Stochastic Neural Networks

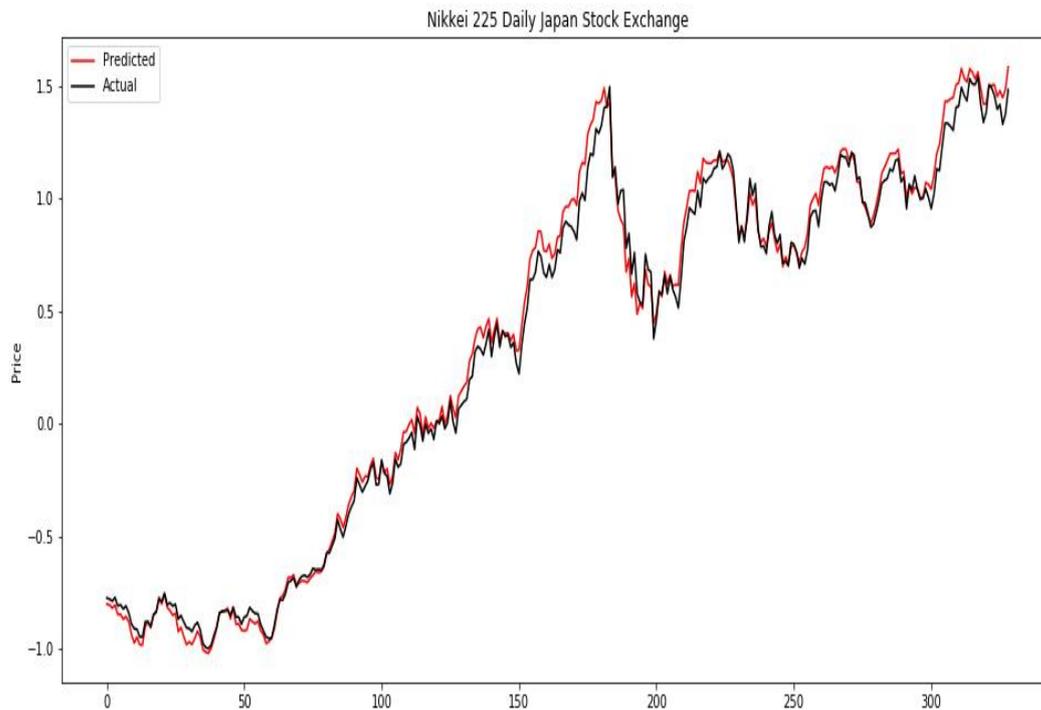


Fig. 3: SNN predicted vs. actual graph for Nikkei 225

Figure 3 shows the pictorial outcome of the predicted Japanese Nikkei 225 Index, which is obtained by simulating the Stochastic Neural Networks model. SNN uses stochastic functions to introduce randomness in the algorithm. The randomness is directly applied to the neurons. This is an internal randomness function, unlike Genetic Algorithms where randomness is obtained

externally. Randomness always helps to escape local minima and search for global minima. It can be observed how precise SNN generates the pattern of predicted output which is more or less similar to the actual data.

The prime reason behind obtaining high precise outcome is the normalization of input data with standard scalar.

Table 1: Price-performance metrics for SNN – Nikkei 225

| Index | MAPE | MAE | MSE | RMSE | R^2 | r_1 |
|------------|--------|----------|------------|----------|--------|--------|
| Nikkei 225 | 1.1019 | 139.0807 | 29982.0193 | 173.1532 | 0.9943 | 0.9982 |

Table 2: Trend performance metrics for SNN – Nikkei 225

| Index | Precision (+) | Precision (-) | Recall (+) | Recall (-) | Accuracy | F_1 Score |
|------------|---------------|---------------|------------|------------|----------|-------------|
| Nikkei 225 | 0.8806 | 0.8351 | 0.7867 | 0.9101 | 0.8537 | 0.8527 |

Table 3: Price prediction for SNN – Nikkei 225

| Index | Actual (t_n) | Predicted (t_n) | Actual (t_{n+1}) | Predicted (t_{n+1}) | Actual (t_{n+2}) | Predicted (t_{n+2}) |
|------------|------------------|---------------------|----------------------|-------------------------|----------------------|-------------------------|
| Nikkei 225 | 15864.44 | 15872.86 | 15784.25 | 16052.49 | 15906.57 | 16159.01 |

Interpretation:

The randomness function of Stochastic neural networks has improved the accuracy or hit ratio for Japanese Nikkei 225 Index historical prices. In an attempt to escape local minima, we have considered a model like SNN in an attempt to achieve a higher hit rate. As we can see, we have compromised with RMSE a little to achieve a higher hit ratio.

We can make the following observations.

1. Randomness, when used internally over the weights, improves hit ratio
2. MAPE value is quite less (1.10%) due to randomness, which is suitable for prediction, i.e., lesser the MAPE better the prediction.

3. The improvement in Hit ratio is due to the ability of the randomness function to detect swift changes in the stock to an extent.
4. The model is giving an accuracy of 85.37%, which is higher compared to Qiu and Song, 2016 study.
5. It is also observed that Accuracy and F_1 score are on par with each other, i.e., 85.37% and 85.27% respectively, which is an indication that algorithm is not subject to bias.
6. The Actual prices and predicted prices have a correlation of 99.82% (r_1) which indicates that actual and predicted prices have a very high correlation and are in par with each other and the proposed SNN model can predict the index price.

• Long Short – Term Memory – Stochastic Neural Networks (LSTM – SNN)

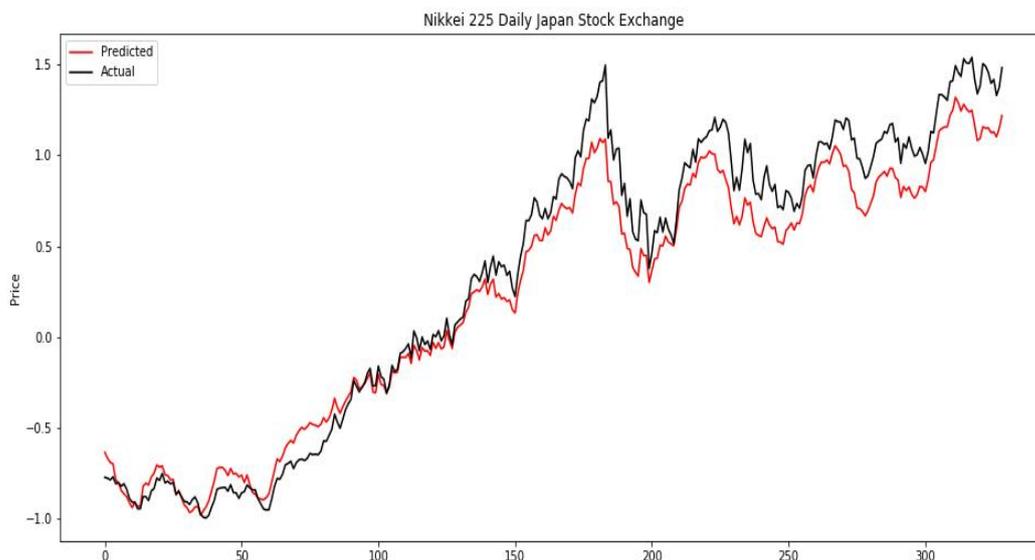


Fig. 4: LSTM-SNN predicted vs. actual graph for Nikkei 225

Figure 4 shows the pictorial outcome of the predicted Japanese Nikkei 225 Index, which is obtained by simulating the fusion of Long-Short Term Memory - Stochastic Neural Networks (LSTM – SNN) model. LSTM - SNN uses LSTM memory function and SNN stochastic functions to introduce

randomness in the algorithm. The randomness is directly applied to the neurons. This is an internal randomness function unlike Genetic Algorithms where randomness is

obtained externally. The memory functions of LSTM helps in storing the patterns in the LSTM and recognizes it when required. Randomness always helps to escape local minima and search for global minima. It can be observed that how precise LSTM - SNN generates the pattern of predicted output which is more or less similar to the actual data. The prime reason behind obtaining high precise outcome is the normalization of input data with a standard scalar

Table 4: Price performance metrics for LSTM-SNN – Nikkei 225

| Index | MAPE | MAE | MSE | RMSE | R^2 | r_1 |
|------------|--------|----------|------------|----------|--------|--------|
| Nikkei 225 | 2.9799 | 387.2129 | 211622.745 | 460.0247 | 0.9598 | 0.9963 |

Table 5: Trend performance metrics for LSTM-SNN – Nikkei 225

| Index | Precision (+) | Precision (-) | Recall (+) | Recall (-) | Accuracy | F ₁ Score |
|------------|---------------|---------------|------------|------------|----------|----------------------|
| Nikkei 225 | 0.8477 | 0.8757 | 0.8533 | 0.8708 | 0.8628 | 0.8628 |

Table 6: Price prediction for LSTM-SNN – Nikkei 225

| Index | Actual (t _n) | Predicted (t _n) | Actual (t _{n+1}) | Predicted (t _{n+1}) | Actual (t _{n+2}) | Predicted (t _{n+2}) |
|------------|--------------------------|-----------------------------|----------------------------|-------------------------------|----------------------------|-------------------------------|
| Nikkei 225 | 15864.44 | 14837.98 | 15784.25 | 15141.07 | 15906.57 | 15198.17 |

Interpretation:

The fusion of memory function and randomness function of Long Short-term Memory and Stochastic neural networks has improved the accuracy or hit ratio for Japanese Nikkei 225 Index historical prices. When we try to pair LSTM with SNN, unlike GA, we observe that advantage introduced by memory function is overwritten by randomness function. This is due to stochastic function directly changing the values within the memory cells of LSTM neurons. Hence, we make two observations here.

1. Randomness, when used internally over the weights, improves hit ratio
2. MAPE value is quite less (2.97%) due to the fusion of memory and randomness function, which is suitable for prediction, i.e., lesser the MAPE better the prediction.

3. The improvement in Hit ratio is due to the ability of the randomness function to detect swift changes in the stock to an extent and memory function to recognise the patterns.
4. The model is giving an accuracy of 86.28%, which is higher compared to Qiu and Song, 2016 study.
5. It is also observed that Accuracy and F₁ score are on par with each other, i.e., 86.28% and 86.28% respectively, which is an indication that algorithm is not subject to bias.
6. The Actual prices and predicted prices have a correlation of 99.63% (r₁) which indicates that actual and predicted prices have a very high correlation and are in par with each other and the proposed hybrid model LSTM - SNN model can predict the index price.

Table 7: Comparison of our study with the prior research report

| Studies | Methods | Stock market | Hit ratio (%) |
|--------------------------------------|---|----------------------------------|---------------|
| de Faria et al. [28] | Adaptive Exponential Smoothing (AES) | Brazilian Stock Exchange | 57% |
| | Artificial Neural Networks (ANN) | | 60% |
| Yu, Wang, and Lai [29] | ARIMA | New York Stock Exchange | 60.71% |
| | FNN | | 63.89% |
| | Support Vector Machines(SVM) | | 64.68% |
| Kara, Acar Boyacioglu and Baykan [4] | Artificial Neural Networks (ANN) | Turkey – Istanbul Stock Exchange | 71.52% |
| | Polynomial Support Vector Machines(SVM) | | 75.74% |
| Dai, Wu, and Lu [30] | Linear Independent Component Analysis – Backpropagation neural network (LICA - BNN) | Shanghai Stock Exchange | 78.26% |
| | Principal Component Analysis (PCA) | | 79.50% |
| Qiu and Song [31] | GA-ANN hybrid model | Japan - Nikkei 225 Index | 81.27 |
| Our study | Stochastic Neural Networks (SNN) | Japan – Nikkei 225 Index | 85.37% |
| | Long Short-Term Memory – Stochastic Neural Networks (LSTM – SNN) | | 86.28% |

It is observed from Table 7 that the proposed models Stochastic Neural Networks and hybrid model Long-Short Term Memory – Stochastic Neural Networks have given better results compared to the previous studies seen in the Table 7. The present study challenged the work of Qiu and Song in the year 2016. Hence the present study also

considered the same index (Japanese Nikkei 225) and the historical prices as considered by Qiu and Song.

B. Discussions

- The daily closing index price is a low momentum trade and more volatile.
- Sometimes when there is a variation in the market due to budget or elections or when the company stock is more volatile in general daily momentum will be more.
- If the algorithm predicts that the price of the stock is going down, then the stock is sold in the morning.
- If the algorithm predicts that the price of the stock is going up, then the stock is bought in the morning.
- If the algorithm has predicted correctly, then investors yield profits otherwise vice – versa.
- It is observed that even if the algorithm has 60% accuracy, profits will be made since there will be more profits than losses.
- The price predictions of the algorithms should be observed in order to make sure when a loss happens; it will not be so much that it will not be able to cover it while making a profit.

V. CONCLUSION AND FUTURE WORK

A. Conclusion

The study aims to introduce Stochastic Neural Network (SNN) for prediction of the Japanese Nikkei 225 Index. The study considered 15 technical indicators in which 11 technical indicators were previously studied by [4] and [15] [16]. The additional four technical indicators considered are On – Balance – Volume (OBV) to measure the volume of the stocks, Bollinger Bands (BB) to evaluate the volatility of the stocks and Price Rate of Change (PROC) and Disparity (DIS) to examine the momentum of the stocks. The evaluation was carried out for historical stock prices data from January 23rd 2007 till December 30th 2013 as considered by Qiu and Song, 2016 [6]. Both the models were able to predict the stock price at accuracy above 85% despite volatility which was higher accuracy compared to Qiu and Song accuracy of 81%. The study proposed a two-stage fusion approach involving Long Short-Term Memory (LSTM) in the first stage. The second stage of the fusion approach uses a Stochastic Neural Network (SNN). The prediction performance of this hybrid model is compared with the single-stage SNN. It is observed from previous studies [15] [16] that two different models were considered for predicting the stock price and the stock trend which led to building a same (single) model for predicting the stock price and trend. The study concludes that the model's accuracy increased with the addition of OBV, ROC, DIS and BB technical indicators when models were tested for Japanese Nikkei 225 Index the models were tested for the same data considered by Qiu and Song, 2016 [6]. It was observed that the present models and selected 15 technical indicators gave higher accuracy than Qiu's research (i.e., 81%). The study concludes that newly introduces technical indicators have a more significant impact on stock market prediction. Through experimental evaluation, it is observed that the randomness function of Stochastic Neural Network gave 85.37% and the fusion of LSTM – SNN gave 86.28%. LSTM - SNN model can be tested for real-time data as it is observed that even if the

algorithm has 60% accuracy, profits will be made since there will be more profits than losses. The present study results are contradicting the major financial theory the Efficient Market Hypothesis (EMH) which claims that the prices of the assets entirely reflect the information which is already available, and it is not possible to defeat the market consistently [3][1]. These models can help the investors to buy and sell their stocks based on the prediction and reduce the risk for the investors.

B. Future work

The study can be explored by taking public sentiments into account as public sentiments impact stock prices significantly. Further, the models can be tested for FOREX trading, derivatives market, commodities, and currency markets. The study was limited to the Japanese Nikkei 225 Index. The research can be extended to different stock market like the London Stock Exchange (LSE), NASDAQ to check for the model's consistency.

ACKNOWLEDGMENT

Acknowledgment goes to M.S Ramaiah University of Applied Sciences, India for having permitted to carry out the investigation and their continuous assistance during the period of the research.

REFERENCES

- [1] B. G. Malkiel, "The efficient market hypothesis and its," *J. Econ. Perspect.*, vol. 17, no. 1, pp. 59–82, 2003.
- [2] A. G. Țițan, "The Efficient Market Hypothesis: Review of Specialized Literature and Empirical Research," *Procedia Econ. Financ.*, vol. 32, no. 15, pp. 442–449, 2015.
- [3] Y. S. Abu-Mostafa and A. F. Atiya, "Introduction to financial forecasting," *Appl. Intell.*, vol. 6, no. 3, pp. 205–213, 1996.
- [4] Y. Kara, M. Acar Boyacioglu, and Ö. K. Baykan, "Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange," *Expert Syst. Appl.*, vol. 38, no. 5, pp. 5311–5319, 2011.
- [5] E. Hadavandi, H. Shavandi, and A. Ghanbari, "Integration of genetic fuzzy systems and artificial neural networks for stock price forecasting," *Knowledge-Based Syst.*, vol. 23, no. 8, pp. 800–808, 2010.
- [6] M. Qiu, Y. Song, and F. Akagi, "Application of artificial neural network for the prediction of stock market returns: The case of the Japanese stock market," *Chaos, Solitons and Fractals*, vol. 85, pp. 1–7, 2016.
- [7] W. Long, Z. Lu, and L. Cui, "Deep learning-based feature engineering for stock price movement prediction," *Knowledge-Based Syst.*, vol. 164, pp. 163–173, 2019.
- [8] R. Singh and S. Srivastava, "Stock prediction using deep learning," *Multimed. Tools Appl.*, vol. 76, no. 18, pp. 18569–18584, 2017.
- [9] E. Chong, C. Han, and F. C. Park, "Deep learning

- networks for stock market analysis and prediction: Methodology, data representations, and case studies,” *Expert Syst. Appl.*, vol. 83, pp. 187–205, 2017.
- [10] G. De Tre, A. Hallez, and A. Bronselaer, “Performance optimization of object comparison,” *Int. J. Intell. Syst.*, vol. 29, no. 2, pp. 495–524, 2014.
- [11] L. Nambiar, D. G. Menon, and A. V. Vidyapeetham, “Predicting Market Prices Using Deep Learning Techniques,” vol. 118, no. 20, pp. 217–223, 2018.
- [12] M. Hagenau, M. Liebmann, and D. Neumann, “Automated news reading: Stock price prediction based on financial news using context-capturing features,” *Decis. Support Syst.*, vol. 55, no. 3, pp. 685–697, 2013.
- [13] C. Evans, K. Pappas, and F. Xhafa, “Utilizing artificial neural networks and genetic algorithms to build an algo-trading model for intra-day foreign exchange speculation,” *Math. Comput. Model.*, vol. 58, no. 5–6, pp. 1249–1266, 2013.
- [14] X. Li et al., “Empirical analysis: stock market prediction via extreme learning machine,” *Neural Comput. Appl.*, vol. 27, no. 1, pp. 67–78, 2016.
- [15] J. Patel, S. Shah, P. Thakkar, and K. Kotecha, “Predicting stock market index using fusion of machine learning techniques,” *Expert Syst. Appl.*, vol. 42, no. 4, pp. 2162–2172, 2015.
- [16] J. Patel, S. Shah, P. Thakkar, and K. Kotecha, “Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning techniques,” *Expert Syst. Appl.*, vol. 42, no. 1, pp. 259–268, 2015.
- [17] A. M. Rather, A. Agarwal, and V. N. Sastry, “Recurrent neural network and a hybrid model for prediction of stock returns,” *Expert Syst. Appl.*, vol. 42, no. 6, pp. 3234–3241, 2015.
- [18] J. Wang and J. Wang, “Forecasting stock market indexes using principle component analysis and stochastic time effective neural networks,” *Neurocomputing*, vol. 156, pp. 68–78, 2015.
- [19] A. H. Moghaddam, M. H. Moghaddam, and M. Esfandyari, “Stock market index prediction using artificial neural network,” *J. Econ. Financ. Adm. Sci.*, vol. 21, no. 41, pp. 89–93, 2016.
- [20] C. L. Cocianu and H. Grigoryan, “Machine learning techniques for stock market prediction. A case study of OMV Petrom,” *Econ. Comput. Econ. Cybern. Stud. Res.*, vol. 50, no. 3, pp. 63–82, 2016.
- [21] L. Di Persio and O. Honchar, “Artificial neural networks architectures for stock price prediction: Comparisons and applications,” *Int. J. Circuits, Syst. Signal Process.*, vol. 10, pp. 403–413, 2016.
- [22] X. Zhong and D. Enke, “Forecasting daily stock market return using dimensionality reduction,” *Expert Syst. Appl.*, vol. 67, pp. 126–139, 2017.
- [23] C. Krauss, X. A. Do, and N. Huck, “Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500,” *Eur. J. Oper. Res.*, vol. 259, no. 2, pp. 689–702, 2017.
- [24] U. M. Mohapatra, B. Majhi, and S. C. Satapathy, “Financial time series prediction using distributed machine learning techniques,” *Neural Comput. Appl.*, pp. 1–16, 2017.
- [25] T. Fischer and C. Krauss, “Deep learning with long short-term memory networks for financial market predictions,” *Eur. J. Oper. Res.*, vol. 270, no. 2, pp. 654–669, 2018.
- [26] Y.-G. Song, Y.-L. Zhou, and R.-J. Han, “Neural networks for stock price prediction,” vol. 00, no. 00, pp. 1–13, 2018.
- [27] M. Zhang, X. Jiang, Z. Fang, Y. Zeng, and K. Xu, “High-order Hidden Markov Model for trend prediction in financial time series,” *Phys. A Stat. Mech. its Appl.*, vol. 517, pp. 1–12, 2019.
- [28] E. L. de Faria, M. P. Albuquerque, J. L. Gonzalez, J. T. P. Cavalcante, and M. P. Albuquerque, “Predicting the Brazilian stock market through neural networks and adaptive exponential smoothing methods,” *Expert Syst. Appl.*, vol. 36, no. 10, pp. 12506–12509, 2009.
- [29] L. Yu, S. Wang, and K. K. Lai, “A neural-network-based nonlinear metamodeling approach to financial time series forecasting,” *Appl. Soft Comput. J.*, vol. 9, no. 2, pp. 563–574, 2009.
- [30] W. Dai, J. Y. Wu, and C. J. Lu, “Combining nonlinear independent component analysis and neural network for the prediction of Asian stock market indexes,” *Expert Syst. Appl.*, vol. 39, no. 4, pp. 4444–4452, 2012.
- [31] M. Qiu and Y. Song, “Predicting the direction of stock market index movement using an optimized artificial neural network model,” *PLoS One*, vol. 11, no. 5, pp. 1–11, 2016.

ABOUT THE AUTHORS

Bhanu Teja Reddy pursued Bachelor of Engineering in Medical Electronics from M.S Ramaiah Institute of Technology, Bengaluru in 2016 and at present pursuing her Master of Business Administration in Financial Management from M.S Ramaiah University of Applied Sciences, Bengaluru from 2017. Her main research work focuses on the Development of Neural Network and Genetic Algorithm for the stock price and trend prediction of selected Indian and U.S.

Usha J.C is pursuing her Ph.D. in Ramaiah University of Applied Sciences, Bangalore India from 2016 onwards. She pursued her Masters in M.Com (Finance) from Mangalore University, Mangalore (2000-2002) and also pursued MBA in Finance from Manipal University, Manipal (2009 – 2011) and her Bachelor’s degree in B.Com, Mangalore University (1998 – 2000). At present, she is working as Assistant Professor in Faculty of Management and Commerce department from 2015 in Ramaiah University of Applied Sciences, Bangalore. She previously worked as Head of the Department Commerce (Yenepoya College) 2011 for two years in Deeksha Ace Creating Learning, Bangalore. She is also an author for the book, Usha JC (2010) Basics of statistics, Indian knowledge corporation publications. She has published more than ten papers in various ICI, Scopus and UGC listed journal