Integration of genetic algorithm with artificial neural network for stock market forecasting

Dinesh K. Sharma¹ · H. S. Hota² · Kate Brown¹ · Richa Handa³

Received: 20 May 2021 / Revised: 15 July 2021 / Accepted: 27 July 2021 / Published online: 17 August 2021

© The Society for Reliability Engineering, Quality and Operations Management (SREQOM), India and The Division of Operation and Maintenance, Lulea University of Technology, Sweden 2021

Abstract Traditional statistical as well as artificial intelligence techniques are widely used for stock market forecasting. Due to the nonlinearity in stock data, a model developed using the traditional or a single intelligent technique may not accurately forecast results. Therefore, there is a need to develop a hybridization of intelligent techniques for an effective predictive model. In this study, we propose an intelligent forecasting method based on a hybrid of an Artificial Neural Network (ANN) and a Genetic Algorithm (GA) and uses two US stock market indices, DOW30 and NASDAQ100, for forecasting. The data were partitioned into training, testing, and validation datasets. The model validation was done on the stock data of the COVID-19 period. The experimental findings obtained using the DOW30 and NASDAQ100 reveal that the accuracy of the GA and ANN hybrid model for the DOW30 and NASDAQ100 is greater than that of the single ANN (BPANN) technique, both in the short and long term.

Keywords Artificial neural networks (ANN) · Genetic algorithms (GA) · Back propagation neural network (BPANN) · Stock market forecasting

1 Introduction

Academics and financial experts are interested in financial forecasting to make stock market pricing predictions. The random behavior of stock markets makes forecasting difficult and new approaches to forecasting models continue to be sought. Traditional statistical techniques such as autoregressive integrated moving average (ARIMA) (Box & Jenkins, 1976), autoregressive conditional heteroscedasticity (ARCH) (Engle, 1982), and generalized autoregressive conditional heteroscedasticity (GARCH) (Bollerslev, 1986) were developed early in stock market forecasting. However, these models are generally not effective tools for forecasting due to the non-linearity of data and the occurrence of shocks (Sharma et al. 2016).

Several authors have worked with various artificial intelligence (AI) techniques for financial time series forecasting in the literature. Others have studied stock market data based on various artificial neural network (ANN) techniques. From the study of this literature, ANNs outperform the traditional techniques with regard to forecasting non-linear time-series data (Aiken & Bsat, 1999; Yao, 1999; Sharma & Rababaah, 2014). Lee (2004) proposed an intelligent model using a hybrid radial basis functional network based on stock market predictions. Several academicians have applied ANN techniques for stock price predictions (Grigoryan, 2015; Guresen et al. 2011; Kazem et al. 2013; Laboissiere et al. 2015; Naeini, et al. 2010; Siew & Nordin, 2012). Lahmiri (2014) proposed a

✉ Dinesh K. Sharma
dksharma@umes.edu

H. S. Hota
proffhota@gmail.com

Kate Brown
kbrown@umes.edu

Richa Handa
richihanda@yahoo.com

¹ University of Maryland Eastern Shore, Princess Anne, USA

² Atal Bihari Vajpayee University, Bilaspur, India

³ D.P. Vipra College, Bilaspur, India

predictive model using the wavelet transform for signal denoising and backpropagation for time series forecasting. Many authors found deep learning is helpful for the prediction of non-linear time series data. Additionally, numerous authors (Ni et al. 2019; Thakkar & Chaudhari, 2020; Yadav et al. 2020; Lin et al. 2021) have proposed a new forecasting method for forecasting foreign exchange and stock market data using deep learning architectures, i.e., recurrent neural network, long short-term memory, and convolutional neural network. Thakkar & Chaudhari (2021) test nine deep neural network (DNN) models for stock market predictions.

Various authors (Huang & Wu, 2008; Jang, 1993; Sharma & Rababaah, 2014; Rababaah & Sharma, 2015; Singh et al. 2020) have also suggested that sometimes a single technique is not sufficient and does not provide accurate results. Sharma and Rababaah (2014) developed a model that forecasted US stock market trends by combining signal processing and ANN. Rababaah & Sharma (2015) enhanced the model even further by combining two distinct signal-processing techniques with ANN. Weng et al. (2018) proposed an intelligent system composed of two modules: a knowledge base and AI, and they extracted new features to improve the model's performance. Nayak & Misra (2019) combined a chemical reaction optimization module to optimize the weights combined with a neuro-fuzzy network (CNFN) to predict stock index returns.

After reviewing the literature, we found that ANNs are good for learning input–output patterns. However, they may face local minima and network paralysis problems due to rough weights assigned by learning algorithms like backpropagation. On the other side, genetic algorithms are good for optimization but difficult to use to find suitable fitness functions. To overcome these drawbacks, we propose a model combining the strengths of two techniques to develop a hybrid GANN model to get more accurate forecasting results. We use GAs to optimize the weights of ANNs so that the results produced by ANNs will be more accurate and the prediction errors will be minimized.

This study has three goals. The first is to design and build a GAAN to forecast stock data. The system is divided into the GA and the ANN modules. The weights of the ANN are optimized using GA, and resulting ANN is used to make predictions. The second is to validate the model with actual stock market data to check the performance of the model. Validation is performed on the model using stock data during COVID-19 from March 1, 2020, to October 8, 2020. The third is to use GANN to predict the Dow30 and NASDAQ100 indices closing prices for the next day. The results and patterns of actual and predicted values for the proposed model are compared to the same data evaluated by a BPANN.

The remainder of the paper is laid out as follows: Sect. 2 provides a brief review of previous studies. The proposed framework methodology is discussed in Sect. 3. The data and experimental work to be used for stock market forecasting are described in Sect. 4. The findings and discussion are presented in Sect. 5. Finally, Sect. 6 provides closing remarks.

2 Review of the literature

In the literature review, many authors (Branke, 1995; Yao, 1999; Mandziuk & Jaruszewicz, 2011; Sermpinis et al. 2015; Alhnaity & Abbod, 2020) have worked with ANNs and GAs for stock market predictions with weight optimization. Pan et al. (2005) demonstrated the efficiency of the proposed method using a mutation-only GA. Mandziuk & Jaruszewicz (2011) proposed a neuro-genetic system for stock market predictions. Many authors (Cai et al. 2013; Kuo et al. 2001) have done financial data forecasting using ANNs, Fuzzy Logic, and GAs. Majhi et al. (2014) and Sermpinis et al. (2015) have also proposed models for forecasting various time series data using several machine learning techniques and GAs and found that the GAs outperformed the machine learning techniques.

Alhnaity & Abbod (2020) proposed a novel hybrid intelligent model for time series prediction using ANNs, support vector regression (SVR), feature extraction, with GAs to optimize weights. Prado et al. (2020) proposed a novel ensemble methodology for forecasting aggregated long-term energy demand that included an ARIMA, ANN, fuzzy inference system model, adaptive neuro-fuzzy inference system, SVR, extreme ML, and GA. Huang et al. (2021) presented a GA-based model for financial data forecasting using VMD and LSTM. Recently, Peng et al. (2021) studied feature selection in the context of DNN models that use technical analysis indicators to predict stock price direction.

A literature survey found that a few papers were focused on the impact of expected or unexpected events that affect the trends of time series data using intelligent techniques. Goodell & Vähämaa (2013) and Sharma et al. (2017) focus on the US presidential election in time series data. Mo et al. (2016) did forecasts for the financial cross-correlation relationship using an Exponent Back-Propagation ANN, while Hota et al. (2018) analyzed the impact of demonetization in the Indian stock market and foreign exchange rates. Kumar & Kumara (2020) utilized pre and post COVID-19 effects in market capitalization. Balli (2020) also proposed data analysis of the pandemic using machine learning techniques like learning regression, support vector machine (SVM), multilayer perceptron, and random forecast.

3 Methodology

In this section, we present a framework of the proposed methodology, which includes a brief overview of ANN and GA concepts, and comprehensive descriptions of Genetically tuned Artificial Neural Network (GANN) system development.

3.1 Artificial neural network

An artificial neural network (Zurada et al. 1994; Rajashekharan & Pai, 1996) is inspired by the human nervous system, such as the brain. Due to fluctuating behavior of financial time series data ANN is used to develop forecasting models that produce results more accurately than statistical models. ANN has a three-layer architecture: Input layer, Hidden layer, and Output layer, which is sufficient to solve a complex nonlinear problem like time series forecasting. An ANN architecture with three layers is shown in Fig. 1. ANN consists of two Phases: Feedforward Neural Network and Feedback Neural Network. An ANN with a single hidden layer feedforward with one output node is commonly used to develop forecasting applications.

3.2 Genetic algorithm

A genetic algorithm (Zurada et al. 1994) is a general-purpose, population-based search algorithm that begins with the population of “genes”, which represent the possible solutions to the considered problem. Strings of values representing a solution to the problem are joined together and referred to as a “Chromosome.” A fitness value is assigned to each string called “fitness value,” which identifies how good a solution it is. The basic cycle of the genetic algorithm is shown in Fig. 2. First, we initialize the population size. Second, the fitness values are calculated using equation number 2. Finally, we use these fitness values to create new generations using three genetic operators: selection, crossover, and mutation.

Fig. 1 A simple architecture of Artificial Neural Network for stock price

3.3 Genetically tuned artificial neural network (GANN)

Local minima and network paralysis are some drawbacks of ANN. The network cannot adjust the weights towards local minima, and the network becomes paralyzed, which affects the system’s accuracy. On the other hand, sometimes we cannot get optimal solutions from the genetic algorithm alone. To overcome these problems, the hybridization of ANN with GA is required to develop a new forecasting model. Figure 3 depicts the process flow diagram and overall scheme of GANN for stock data forecasting. Each of the components of this figure is described as below:

3.3.1 Data set

We have collected two stock datasets: DOW30 and NASDAQ100, from www.yahoofinance.com as summarized in Table 1. After data collection, normalization was used by scaling the data in the range of [0 1] to simplify the learning process (Nawi et al., 2013; Sola & Sevilla, 1997) and to improve the accuracy of predictive model as shown in Eq. 1:

$$X_{new} = \frac{X}{X_{max}} \quad (1)$$

where x is observed value of the time series data, X_{max} is highest value of observation of a particular feature while X_{new} is the calculated normalized observation.

3.3.2 Data partition

The sample size of partitioned data must be optimum because the result of ANN based system depends on the training and testing step and the accuracy of the model may vary based on this data partition. To choose the best sample size for a particular model and particular data, the index data is divided into two different partitions (90%-10% and 80%-20%). In this proposed work we have downloaded DOW30 and NASDAQ100 data sets from March 01, 2017 to Oct 08, 2020. The data is divided into three parts: training, testing and validation. Each partition of data is tuned with GANN to get the forecasting result. The validation data set consist of data from March 01, 2020 to October 08, 2020, i.e. the period of COVID-19 (155 samples). During the pandemic, stock data became volatile and uncertainty about many of the foundations of the economy increased. We tested the value of the model by seeing how accurately it performed while working with the COVID-19 dataset.

Fig. 2 Basic Cycle of Genetic algorithm **Fig. 3** Scheme of GA tuned ANN (GANN)**Table 1** Summary of Data

Particular	Detail
Index data	DOW30 and NASDAQ100
Periods	March 01, 2017 to Oct 08, 2020 (3 Years)
Total # of samples	910
Downloaded from	http://www.yahoofinance.com
Data partition	Training, Testing, and Validation

3.3.3 Initial population and coding

The initial population is required to work with the GA. Potential solutions to the problem called genes are joined together to form a string of values referred to as chromosomes.

3.3.4 Fitness function

The fitness function is used to calculate the fitness value by assigning merits to each individual in population as shown in Eq. 2.

$$F = \frac{1}{E} \quad (2)$$

where E is the root mean square of the error as shown below in Eq. 3.

$$E = \sqrt{\frac{(E_1 + E_2 + E_3 + \dots + E_n)}{n}} \quad (3)$$

3.3.5 Selection

The selection of the best parents, which mate and recombine to create offspring for the next generation is a crucial task to do in GA. Chromosomes selected from the population to be parents will produce offspring. In this study, selection of superior parents was done with a Roulette wheel selection operator. In Roulette wheel selection, the circular wheel is divided by the available chromosomes and a fixed point on the wheel is set. Whichever area of the wheel comes in front of this fixed point is chosen as a parent, and the same process is repeated for the other parent.

3.3.7 Mutation

This is an optional operator in which a small change in chromosomes has been allowed to get something new to the individual with very low mutation probability. We have applied a two-point mutation operator in the evolution process to optimize the weights of the ANN.

3.3.8 Replace

In this method, the worst fit chromosome is replaced with best fit chromosome to get better set for next generation.

3.3.9 Weight extraction

Weights must be extracted from each of the chromosomes to calculate the fitness value. Let $x_1, x_2, \dots, x_d, \dots, x_L$ represent a chromosome and $x_{kd+1}, x_{kd+2}, \dots, x_{(k+1)d}$ represent the k^{th} gene ($k \geq 0$) in the chromosome. Then the weight extraction formula is given by the following equation:

$$W_k = \begin{cases} + \frac{X_{kd+2}10^{d-2} + X_{kd+3}10^{d-3} + \dots + X_{(k+1)d}}{10^{d-2}} & \text{if } 5 \leq X_{kd+1} \leq 9 \\ - \frac{X_{kd+2}10^{d-2} + X_{kd+3}10^{d-3} + \dots + X_{(k+1)d}}{10^{d-2}} & \text{if } 0 \leq X_{kd+1} \leq 5 \end{cases} \quad (4)$$

3.3.6 Crossover

In crossover, two parents are selected to produce one or more offspring along with genetic characteristics of the parents. In this proposed work, multi-point crossover is used, where alternate segments are swapped to get new offspring as shown in Fig. 4.

3.3.9.1 Evaluation criteria Various error metrics have been used to assess the efficiency of models. We assume Mean Absolute Percentage Error (MAPE), Mean Square Error (MSE), and Root Mean Square Error (RMSE) in this proposed study. The following are the formulas for these measures:

Mean Absolute Percentage Error (MAPE):

$$MAPE = \frac{1}{N} \left[\sum_{i=1}^N \frac{|AV_i - PV_i|}{AV_i} \times 100 \right] \quad (5)$$

Mean Square Error (MSE):

Fig. 4 Two point crossover (Real coded genes)

$$MSE = \frac{1}{N} \sum_{i=1}^N (AV_i - PV_i)^2 \quad (6)$$

Root Mean Square Error (RMSE):

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (AV_i - PV_i)^2} \quad (7)$$

where AV_i is actual value for i th sample and PV_i is predicted value for i th sample and N is the total number of samples.

4 Data and experimental work

In this section, we present the data description of two US stock indices and the experimental work for stock market forecasting with the GANN compared to a backpropagating ANN (BPANN).

4.1 Data description

To develop the predictive model we have used two stock indices i.e. DOW30 and NASDAQ100, collected from www.yahoofinance.com. We have collected data from March 01, 2017 to October 08, 2020 that covers three years of data with these features: Open, High, Low and Close; and Next Day Close is used as the output data. The data we have used in this proposed work depicted in Table 1.

Trends of historical data of the DOW30 and NASDAQ100 are shown in Fig. 5 (a) and (b) which show that the stock data we have collected is non-linear in nature. Based on the literature, statistical techniques may not be able to develop effective predictive models for data with fluctuating behavior. In this work, we have proposed a

Genetically tuned ANN model to develop a forecasting model for this data.

4.2 Experimental work

The experimental work is done with the model developed and discussed above using MATLAB (MathWorks, 2018). The experimental work and the results are explained in the following five subsections; experiment with DOW30 data set, experiment with NASDAQ100 data set, error measures, model selection, and N-Days ahead forecasting. The study investigated the GANN's performance compared to BPANN using all three partitions of both data sets. The system training took place using each partition with the GA to optimize the weights and independently by the BPANN. After training, the model was tested for prediction, model validation was done to check the performance of the model, and predictions were made for the next day's closing prices.

4.2.1 Experiment with DOW30 data set

Both the models were trained and then tested with both partitions one by one. Comparative graphs showing the actual and predicted closing price by GANN at the testing stage for both partitions are shown in Fig. 6 (a) and (b).

4.2.2 Experiment with NASDAQ100 data set

A similar experiment was also carried out for the NASDAQ100 data set for both partitions, and the actual value and predicted value for these partitions are shown with the graphs in Fig. 7 (a) and (b). It can be clearly observed from these graphs that the predicted value is very near to the

Fig. 5 Graphical Trend of **a** DOW30 and **b** NASDAQ100 Data

Fig. 6 Actual vs Predicted DOW30 output using GANN at testing stage **a** Partition 1, **b** Partition 2

actual value. Hence, we can say that GANN is performing well for the index data set.

4.2.3 Error measures

Three different error measures: MAPE, MSE, and RMSE, are calculated using Eqs. 5, 6, and 7 respectively after simulations of BPANN and GANN with both the partitions of the DOW30 data set, and the results are tabulated in Tables 2, 3, and 4 for the training, testing and validation

datasets for MAPE, MSE and RMSE respectively. We can observe from these tables that we get consistent MAPE values in the context of partition size and the model we have used. Training error is less than testing error for all the partitions, while validation error is higher. In the case of the NASDAQ100 data set, the same pattern can be observed in Tables 5, 6, and 7 for MAPE, MSE, and RMSE, respectively. The range of MAPE, in this case, is higher than DOW30. This may be due to the characteristics of the stocks making up the two indices. For almost all the

Fig. 7 Actual Vs Predicted output for NASDAQ100 using GANN at testing stage **a** Partition 1, **b** Partition 2

Table 2 Comparative MAPE of DOW30

Data partition	DOW30					
	BPANN			GANN		
	Training	Testing	Validation	Training	Testing	Validation
Partition 1(90–10)	0.6115	1.5419	2	0.5827	0.8621	1.8731
Partition 2(80–20)	0.5764	1.3477	1.9622	0.5648	0.8268	1.856

Table 3 Comparative MSE of DOW30

Data partition	DOW30					
	BPANN			GANN		
	Training	Testing	Validation	Training	Testing	Validation
Partition 1(90–10)	0.0051	0.0149	0.0164	0.0049	0.0082	0.0153
Partition 2(80–20)	0.0048	0.0087	0.0161	0.0047	0.0077	0.0152

Table 4 Comparative RMSE of DOW30

Data partition	DOW30					
	BPANN			GANN		
	Training	Testing	Validation	Training	Testing	Validation
Partition 1(90–10)	0.0716	0.1219	0.1289	0.0697	0.0908	0.1337
Partition 2(80–20)	0.069	0.0932	0.1285	0.0683	0.0676	0.1269

Table 5 Comparative MAPE of NASDAQ100

Data partition	NASDAQ100					
	BPANN			GANN		
	Training	Testing	Validation	Training	Testing	Validation
Partition 1(90–10)	0.7191	1.3201	5.4307	0.7191	1.3201	5.4307
Partition 2(80–20)	0.758	1.3032	4.723	0.7187	0.879	4.5806

Table 6 Comparative MSE of NASDAQ100

Data partition	NASDAQ100					
	BPANN			GANN		
	Training	Testing	Validation	Training	Testing	Validation
Partition 1(90–10)	0.0043	0.0101	0.0464	0.0044	0.0074	0.0461
Partition 2(80–20)	0.0045	0.0094	0.0403	0.0043	0.0062	0.0392

Table 7 Comparative RMSE of NASDAQ100

Data partition	NASDAQ100					
	BPANN			GANN		
	Training	Testing	Validation	Training	Testing	Validation
Partition 1(90–10)	0.068	0.1004	0.2154	0.066	0.0858	0.191
Partition 2(80–20)	0.067	0.0971	0.2008	0.0653	0.0787	0.168

partitions for both datasets, GANN outperformed the traditional BPANN model as represented by lower errors.

5 Results and discussions

A comparative result based on testing data set for all the three error measures: MAPE, MSE, and RMSE for both the indices is depicted in Fig. 8 (a)–(f), which clearly show the higher performance of GANN for stock index forecasting. Although the range of error in case of GANN is very close to that of BPANN, it is lower for both the partitions and for both the indices, which confirms that hybrid techniques can produce better results by avoiding the weakness of individual techniques.

5.1 Model selection

The best model we have selected based on the error measures shown in Eqs. 5, 6, and 7 is GANN. Partition 2

(80–20) of DOW30 resulted in the most precise prediction for the next day's close prices; hence, a model trained with partition 2 was selected to make N days ahead predictions. A similar experiment took place for the NASDAQ100 data sets, and partition 2(80–20) yielded the best prediction for the next day's close price compared to other partitions. Therefore, this partition was selected to make N days ahead predictions. ANN is trained with partition 2 of DOW30 data set and NASDAQ100 data set with the help of GA, and finally, weights are optimized as shown in Tables 8 and 9 respectively for DOW30 and NASDAQ100 data sets.

After performing various iterations, ANN has converged towards the global optimum due to the chromosomes' final set for the DOW30 and NASDAQ100 data sets. We can observe from these tables that all the chromosomes in the final population have the same genetic material (genes), which shows the GANN's ability and that ANN can optimize the weights through GA. Weights obtained for different input to hidden layer and from hidden to the output layer along with the bias weights are tabulated in Table 8 in

Fig. 8 A comparative bar chart of two models based on testing data for **a** MAPE of DOW30 **b** MAPE of NASDAQ100 **c** MSE of DOW30 **d** MSE of NASDAQ100 **e** RMSE of DOW30 **f** RMSE of NASDAQ100

Table 8 Final Weights of ANN optimized by GA for Partition 2 of DOW30 Data Set

Input to hidden layer (From all the neurons of the input layer to jth neuron of hidden layer)	Bias to hidden layer	Hidden to output layer (From all the neurons of the hidden layer to kth neuron of the outer layer)	Bias to the output layer
−0.6892 1.9630 0.6707 −0.3185	2.4486	0.5918	0.4163
−0.0278 −0.2867 −1.3888 2.5490	0.0658	0.7678	—
−1.7481 0.4978 −0.6309 1.5317	0.0938	−0.5615	—
−2.3880 −0.3028 1.8200 1.5305	−1.412	1.1214	—

the case of partition 2 for the DOW30 data set and in Table 9 for the NASDAQ100 data set. These weights are extracted from the final population's chromosomes using Eq. 4 for the DOW30 and NASDAQ100 data sets.

5.2 N-Days ahead forecasting

To demonstrate and validate the GANN model, a mathematical calculation was performed at the testing stage, which explains the process of N-days ahead forecasting by

Table 9 Final Weights of ANN optimized by GA for Partition 2 of NASDAQ100 Data Set

Input to hidden layer (From all the neurons of the input layer to jth neuron of the hidden layer)	Bias to the hidden layer	Hidden to output layer (From all the neurons of the hidden layer to kth neuron of the outer layer)	Bias to output the layer
0.2082 −0.9984 1.7412 0.1413	−1.6325	0.6354	0.6477
1.0107 −0.0961 1.3453 −0.0807	−1.251	−0.0112	−
−0.0259 −0.0625 −0.0580 −0.4037	−0.3183	−1.7445	−
1.4139 −0.4771 0.1886 −1.4067	1.6475	−0.6425	−

Table 10 Error Measures for N-Days ahead Forecasting for DOW30 data set at testing stage using ANN and GANN for Best Partition (80–20)

No. of day(s)	BPANN			GANN		
	MAPE	MSE	RMSE	MAPE	MSE	RMSE
5	3.904	0.029	0.217	2.255	0.021	0.145
10	5.098	0.049	0.231	3.790	0.036	0.189
20	7.187	0.072	0.308	6.58	0.062	0.249
30	12.428	0.109	0.390	8.195	0.077	0.278

Table 11 Error Measures for N-Days ahead Forecasting for NASDAQ100 data set at testing stage using ANN and GANN for Best Partition (80–20)

No. of day(s)	BPANN			GANN		
	MAPE	MSE	RMSE	MAPE	MSE	RMSE
5	4.091	0.029	0.267	2.841	0.020	0.143
10	6.011	0.052	0.398	5.567	0.040	0.201
20	7.909	0.067	0.445	7.332	0.054	0.233
30	11.876	0.089	0.489	8.819	0.065	0.255

GANN. Weights of the best partition 2 for both DOW30 and NASDAQ100 as shown in Tables 8 and 9 are considered here for the calculation. The study further investigated BPANN's and GANN's comparative capabilities to predict future prices ranging from one day to 30 days. Also, the prediction accuracy was assessed on the future term (number of days) forecasting. The study started to accomplish this task from day five and gradually incremented up to thirty days. N-days ahead prediction for the DOW30 and NASDAQ100 datasets for $N = 5, 10, 20$, and 30 days are calculated. GANN outperforms BPANN for all measures in the N-Day forecasts for both indices.

For each experiment on N-Days forecasting for the DOW30 and NASDAQ100 datasets, the three error measures MAPE, MSE, and RMSE were calculated and presented in Tables 10 and 11, and shown in Figs. 9 and 10. From the Tables, the GANN model's accuracy for 5-days ahead forecasting was 97.75 percent for the DOW30 dataset while 97.16 percent for the NASDAQ100 dataset. The pattern of lower errors for the GANN model continued for 10, 20, and 30 days forecasts. For both indices, the results indicated that GANN outperformed BPANN for short and long-term index predictions. Consistent with expectations, the results demonstrate that moving from short to long term forecasting reduces performance and increases uncertainty. Regardless, the GANN models

Fig. 9 Comparative bar chart for N-Days ahead forecasting between BPANN and GANN for DOW30

Fig. 10 Comparative bar chart for N-Days ahead forecasting between BPANN and GANN for NASDAQ100

performed better than the BPANN even when tested on the pandemic period data.

6 Conclusions

In this study, we proposed an intelligent hybrid framework for stock forecasting. Back Propagation Neural Network (BPANN) and a combination of ANN and GA were used as intelligent techniques to construct forecasting models using the DOW30 and NASDAQ100 datasets. To test the model's efficacy, the data sets were partitioned into three separate partitions: training, testing, and validation. The model was validated using market data during COVID-19, which ran from March 1, 2020 to October 8, 2020, to see how accurately the model performs in the face of any unexpected event, when the stock market was highly volatile. It was discovered that GANN offered more reliable predictive results than the BPANN model for almost all of the data partitions. The performance of both models was tested using a variety of error tests, including MAPE, MSE, and RMSE. When using the principle of global optimization, GANN performed well, but BPANN failed to optimize the weights. This demonstrates that GANN can capture the fluctuating behavior of stock data more intelligently than the BPANN alone. Even though data was collected before and during COVID-19, the accuracy of the GANN model for 5-Days ahead forecasting was 97.75 percent for the DOW30 dataset and 97.16 percent for the NASDAQ100 dataset. Moving from short to long term forecasting reduces performance for both models, but the GANN consistently produces smaller errors.

The future scope of our proposed work can be carried out using some new hybrid techniques that incorporate wavelet transform, feature extraction, feature selection, Adaptive Neuro Fuzzy Inference, or wavelet techniques with Genetic Algorithm. To improve the accuracy of the

results, various tuning parameters such as learning rate, momentum, and so on can be combined with GANN.

Funding We have not received funding for this research.

Declarations

Conflicts of interest The authors have reported no conflicts of interest.

Human and animal rights The authors declared that this article contains no studies involving human subjects or animals.

References

- Aiken M, Bsat M (1999) Forecasting market trends with neural networks. *Inf Syst Manag* 16(4):42–48
- Alhnaity B, Abbod M (2020) A new hybrid financial time series prediction model. *Eng Appl Artif Intell* 95:103873. <https://doi.org/10.1016/j.engappai.2020.103873>
- Balli S (2020) Data analysis of Covid-19 pandemic and short-term cumulative case forecasting using machine learning time series models. *Chaos Solitons Fractals Interdiscip J Nonlinear Sci Nonequilib Complex Phenomena*. <https://doi.org/10.1016/j.chaos.2020.110512>
- Bollerslev T (1986) Generalized autoregressive conditional heteroskedasticity. *J Econom* 31:307–327
- Box G, Jenkins G (1976) *Time series analysis: forecasting and control*. Holden-Day, San Francisco
- Branke J (1995) *Evolutionary algorithms for neural network design and training*. Univ, Karlsruhe, Inst. AIFB, Karlsruhe, Germany, Tech. Rep, p 322
- Cai QS, Zhang D, Wu B, Leung SCH (2013) A novel stock forecasting model based on fuzzy time series and genetic algorithm. *Proc Comput Sci* 18:1155–1162. <https://doi.org/10.1016/j.procs.2013.05.281>
- Engle R (1982) Autoregressive conditional heteroskedasticity with estimates of the variance of United Kingdom inflation. *Econometrica* 50:987–1007
- Goodell JW, Vähämaa S (2013) US presidential elections and implied volatility: the role of political uncertainty. *J Bank Finance* 37(3):1108–1117. <https://doi.org/10.1016/j.jbankfin.2012.12.001>

- Grigoryan H (2015) Stock market prediction using artificial neural networks case study of TALIT, nasdaq OMX baltic stock. *Database Syst J* 6(2):14–23
- Guresen E, Kayakutlu G, Daim TU (2011) Using artificial neural network models in stock market index prediction. *Expert Syst Appl* 38(8):10389–10397. <https://doi.org/10.1016/j.eswa.2011.02.068>
- Hota HS, Handa R, Shrivastava AK (2018) Neural network techniques to develop a robust financial time series forecasting system. *Int J Pure Appl Math* 118(19):125–133
- Huang SC, Wu TK (2008) Integrating GA-based time-scale feature extractions with SVMs for stock index forecasting. *Expert Syst Appl* 35(4):2080–2088. <https://doi.org/10.1016/j.eswa.2007.09.027>
- Huang Y, Gao Y, Gan Y, Ye M (2021) A new financial data forecasting model using genetic algorithm and long short-term memory network. *Neurocomputing* 425:207–218. <https://doi.org/10.1016/j.neucom.2020.04.086>
- Jang JSR (1993) ANFIS: adaptive-network-based fuzzy inference system. *IEEE Trans Syst Man Cybern* 23(3):665–685. <https://doi.org/10.1109/21.256541>
- Kazem A, Sharifi E, Hussain FK, Saberi M, Hussain OK (2013) Support vector regression with chaos-based firefly algorithm for stock market price forecasting. *Appl Soft Comput J* 13(2):947–958. <https://doi.org/10.1016/j.asoc.2012.09.024>
- Kumar MP, Kumara NVM (2020) Market capitalization: pre and post COVID-19 analysis. *Mater Today Proc* 37(2):2553–2557. <https://doi.org/10.1016/j.matpr.2020.08.493>
- Kuo RJ, Chen CH, Hwang YC (2001) An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network. *Fuzzy Sets Syst* 118(1):21–45. [https://doi.org/10.1016/S0165-0114\(98\)00399-6](https://doi.org/10.1016/S0165-0114(98)00399-6)
- Laboissiere LA, Fernandes RAS, Lage GG (2015) Maximum and minimum stock price forecasting of Brazilian power distribution companies based on artificial neural networks. *Appl Soft Comput J* 35:66–74. <https://doi.org/10.1016/j.asoc.2015.06.005>
- Lahmiri S (2014) Wavelet low- and high-frequency components as features for predicting stock prices with backpropagation neural networks. *J King Saud Univ Comput Inf Sci* 26(2):218–227. <https://doi.org/10.1016/j.jksuci.2013.12.001>
- Lee R (2004) IJADE stock advisor: an intelligent agent based stock prediction system using hybrid RBF recurrent network. *Proc IEEE Trans Syst Man Cybern Part A Syst Hum* 34(3):421–428. <https://doi.org/10.1109/TSMCA.2004.824871>
- Lin Y, Yan Y, Xu J, Liao Y, Ma F (2021) Forecasting stock index price using the CEEMDAN-LSTM model. *North Am J Econ Finance* 57:1–14. <https://doi.org/10.1016/j.najef.2021.101421>
- Majhi B, Rout M, Baghel V (2014) On the development and performance evaluation of a multiobjective GA-based RBF adaptive model for the prediction of stock indices. *J King Saud Univ Comput Inf Sci* 26(3):319–331. <https://doi.org/10.1016/j.jksuci.2013.12.005>
- Mandziuk J, Jaruszewicz M (2011) Neuro-genetic system for stock index prediction. *J Intell Fuzzy* 22(2–3):93–123
- Mo H, Wang J, Niu H (2016) Exponent back propagation neural network forecasting for financial cross-correlation relationship. *Expert Syst Appl* 53:106–116. <https://doi.org/10.1016/j.eswa.2015.12.045>
- Naeini MP, Taremi H, Hashemi HB (2010) Stock market value prediction using neural networks. In: 2010 International conference on computer information systems and industrial management applications, CISIM 2010. pp 132–136. <https://doi.org/10.1109/CISIM.2010.5643675>
- Nawi NM, Atomi WH, Rehman MZ (2013) The effect of data pre-processing on optimized training of artificial neural networks. *Proc Technol* 11:32–39. <https://doi.org/10.1016/j.protcy.2013.12.159>
- Nayak S, Misra B (2019) A chemical-reaction-optimization-based neuro-fuzzy hybrid network for stock closing price prediction. *Financ Innov*. <https://doi.org/10.1186/s40854-019-0153-1>
- Ni L, Li Y, Wang X, Zhang J, Yu J, Qi C (2019) Forecasting of forex time series data based on deep learning. *Proc Comput Sci* 147:647–652. <https://doi.org/10.1016/j.procs.2019.01.189>
- Pan X, Zhang J, Szeto KY (2005) Application of mutation only genetic algorithm for the extraction of investment strategy in financial time series. In: Proceedings of the international conference on neural networks and brain, Beijing, China, pp 1682–1686
- Peng Y, Albuquerque PHM, Kimura H, Saavedra CAPB (2021) Feature selection and deep neural networks for stock price direction forecasting using technical analysis indicators. *Mach Learn Appl* 5:100060. <https://doi.org/10.1016/j.mlwa.2021.100060>
- Prado F, Minutolo MC, Kristjanpoller W (2020) Forecasting based on an ensemble autoregressive moving average-adaptive neuro-fuzzy inference system-neural network-genetic algorithm framework. *Energy* 197:117159. <https://doi.org/10.1016/j.energy.2020.117159>
- Rababaah A, Sharma DK (2015) Integration of two different signal processing techniques with artificial neural network for stock market forecasting. *J Manag Inf Decis Sci* 18(2):63–80
- Rajasekaran S, Pai GAV (1996) Genetic algorithm based weight determination for backpropagation network. In: Proceedings of the fourth international conference on advanced computing, pp 73–79
- Serpinin G, Stasinakis C, Theofilatos K, Karathanasopoulos A (2015) Modeling, forecasting and trading the EUR exchange rates with hybrid rolling genetic algorithms—support vector regression forecast combinations. *Eur J Oper Res* 247(3):831–846. <https://doi.org/10.1016/j.ejor.2015.06.052>
- Sharma DK, Rababaah A (2014) Stock market predictive model based on integration of signal processing and artificial neural network. *Acad Inf Manag Sci J* 17(1):51–70
- Sharma H, Sharma DK, Hota HS (2016) A hybrid neuro-fuzzy model for foreign exchange rate prediction. *Acad Account Financ Stud J* 20(3):1–13
- Sharma DK, Hota HS, Handa R (2017) Analysis of US stock market post presidential election performance using ANN techniques. In: Paper presented at the 2017 annual meeting of the decision sciences institute, Washington, DC, November 18–20, 2017
- Siew HL, Nordin MJ (2012) Regression techniques for the prediction of stock price trend. In: ICSSBE 201-Proceedings, 2012 international conference on statistics in science, business and engineering: empowering decision making with statistical sciences, pp 99–103. <https://doi.org/10.1109/ICSSBE.2012.6396535>
- Singh S, Parmar KS, Kumar J, Makkhan SJS (2020) Development of new hybrid model of discrete wavelet decomposition and autoregressive integrated moving average (ARIMA) models in application to one month forecast the casualties cases of COVID-19. *Chaos Solitons Fractals* 135:1–8. <https://doi.org/10.1016/j.chaos.2020.109866>
- Sola J, Sevilla J (1997) Importance of input data normalization for the application of neural networks to complex industrial problems. *IEEE Trans Nucl Sci* 44(3):1464–1468. <https://doi.org/10.1109/23.589532>
- Thakkar A, Chaudhari K (2020) CREST: cross-reference to exchange-based stock trend prediction using long short-term memory. *Proc Comput Sci* 167(2019):616–625. <https://doi.org/10.1016/j.procs.2020.03.328>

- Thakkar A, Chaudhari K (2021) A comprehensive survey on deep neural networks for stock market: the need, challenges and future directions. *Expert Syst Appl* 177:1–17. <https://doi.org/10.1016/j.eswa.2021.114800>
- Weng B, Lu L, Wang X, Megahed FM, Martinez W (2018) Predicting short-term stock prices using ensemble methods and online data sources. *Expert Syst Appl* 112:258–273. <https://doi.org/10.1016/j.eswa.2018.06.016>
- MathWorks Documentation Center (2018). www.mathworks.com. [Online]. <https://in.mathworks.com/help/fuzzy/what-is-fuzzy-logic.html>. Accessed 13 Sept 2020
- Yadav A, Jha CK, Sharan A (2020) Optimizing LSTM for time series prediction in Indian stock market. *Proc Comput Sci* 167(2019):2091–2100. <https://doi.org/10.1016/j.procs.2020.03.257>
- Yao X (1999) Evolving artificial neural networks. *Proc IEEE* 87(9):1423–1447
- Zurada JM, Marks RJ, Robinson C (eds) (1994) *Computational intelligence: imitating life*. IEEE Press, New York

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.