UMT PRESS

Journal of Mathematical Sciences and Informatics
Journal Homepage: https://journal.umt.edu.my/index.php/jmsi
eISSN: 2948-3697
DOI: http://doi.org/10.46754/jmsi.2024.06.006

REVIEW ON STOCHASTIC HYBRIDISATION OF FEEDFORWARD NEURAL NETWORK IN STOCK MARKET

ASSUNTA MALAR PATRICK VINCENT AND HASSILAH SALLEH*

Faculty of Computer Science and Mathematics, Universiti Malaysia Terengganu, 21030 Kuala Nerus, Terengganu, Malaysia.

Corresponding author: hassilah@umt.edu.my

ARTICLE INFO

Article History:
Received 5 DECEMBER 2022
Accepted 12 FEBRUARY 2024
Published 15 JUNE 2024

Keywords:
Multilayer perceptron; Hybridised neural network; Stochastic neural network; Stochastic time effective neural network; Forecasting financial time series model.

ABSTRACT

The stock market is an example of a stochastic environment in the real world. So, obtaining accurate forecasting models of the stock market can be challenging due to its complex characteristics (noisy environment), which result in uncertainty. Although machine learning models have been widely applied to forecast the market, they fail to capture the presence of stochasticity. As a result, a few studies had proposed a hybridisation of Multilayer Perceptron and stochastic processes. Hence, this review paper aims to provide a systematic review of these hybridised models, which have been obtained from the scientific databases Scopus and Web of Science. Finally, it was found out that only eight studies had been conducted to forecast the stock market with Stochastic Neural Network (SNN), and all of them concluded that it has better accuracy than the deterministic model. Thus, the development of SNN is worth exploring in the future as there is room to explore crossing disciplines between neural networks and stochastic processes to improve forecasting accuracy.

2020 Mathematics Subject Classification:

Introduction

Financial Markets are one of the captivating inventions in this modern era. Johnson *et al*. [1] stated that it is an example of a complicated system in the real world. Stock markets are frequently associated with the financial market, which is influenced by various factors. Apart from that, Giles *et al*. [2] have categorised them into two main categories: Political and macroeconomic environments. In addition to that, Ji *et al*. [3] state that market conditions, major social and economic events, investors' preferences, and companies' managerial decisions are among the other factors that affect the stock market. Hence, it is classified as a high risk, high return, and flexible trading product. Moreover, the random events (influential factors) create a noisy environment in the stock market and result in uncertainty (called stochastic). This complex characteristic of the stock market makes it challenging to predict its future price [4-6]. Therefore, researchers are expanding their horizons in search of a better model that can give an accurate prediction.

Moving on, the stock market can be forecasted using statistical models or machine learning models [7-11]. Moving Average models, exponential smoothing models, decomposition models, and Box-Jenkins models are among the statistical models [9], [10]. Setyawati *et al*. [9] also addressed that Autoregressive Moving Average (ARIMA) is the most popular and powerful time-series model applied to forecasting time series data in finance. Besides that, neural networks, genetic algorithms, fuzzy models, and Support Vector Machines (SVM) are among the models which are categorised as machine learning methods.

Various studies concluded that models based on a statistical method do not perform efficiently as the machine learning models [12-17]. This is because statistical models fail to capture the non-linear pattern present in data [15]. In addition, a study concluded that Artificial Neural Network (ANN) outperforms ARIMA when the author forecasted the stock price of the Dell Stock Index from the New York Stock Exchange [17]. Figure 1 depicts the model's predicted outcome as well as its actual price. From the comparative figure, it can be observed that the forecasted value using ARIMA, and ANN is closer to the actual stock price. However, the forecasted value with ARIMA results in a linear line, and the stock price forecasted with ANN follows the fluctuation of the actual stock price. Besides, another study on financial forecasting by Lahane [18] also concluded that ANN performs better for value forecasting, whereas ARIMA outperforms directional forecasting. It is further supported by Hiransha *et al*. [19], where the authors concluded that neural network outperforms ANN as it fails to identify the non-linearity that exists within the data. Thus, these studies are evidence of the higher accuracy of forecasting with neural networks because the non-linear trend of the historical data can be identified by the neural network.

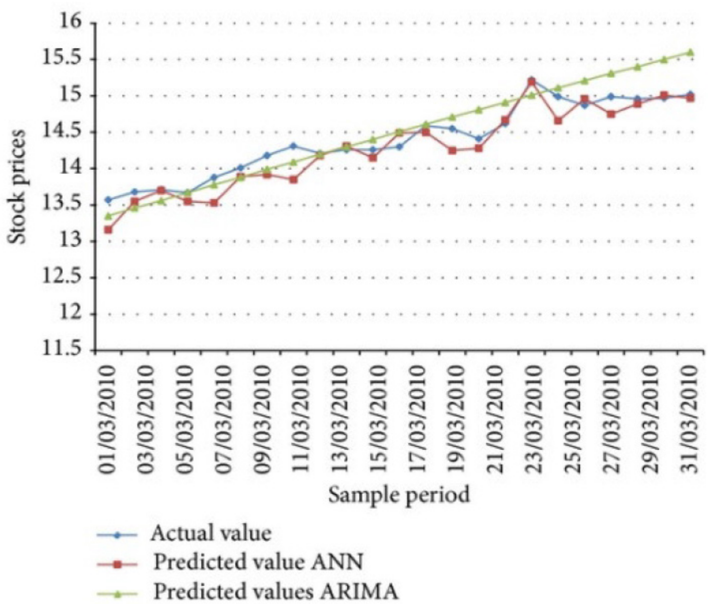

Figure 1: Graph of predicted value of ARIMA and ANN model against actual stock price [17]

Machine learning models can be classified into supervised and unsupervised learning models. The most popular machine learning method applied to forecasting stock prices is supervised learning. A general workflow of supervised learning-based machine learning to forecast stock price is shown in Figure 2 [20]. There are various machine learning models used to forecast stock prices, with ANN, SVM, and their variants being the most common [20, 21] as the forecasted results are promising and extremely effective. Nevertheless, these algorithms are continuously evolving.

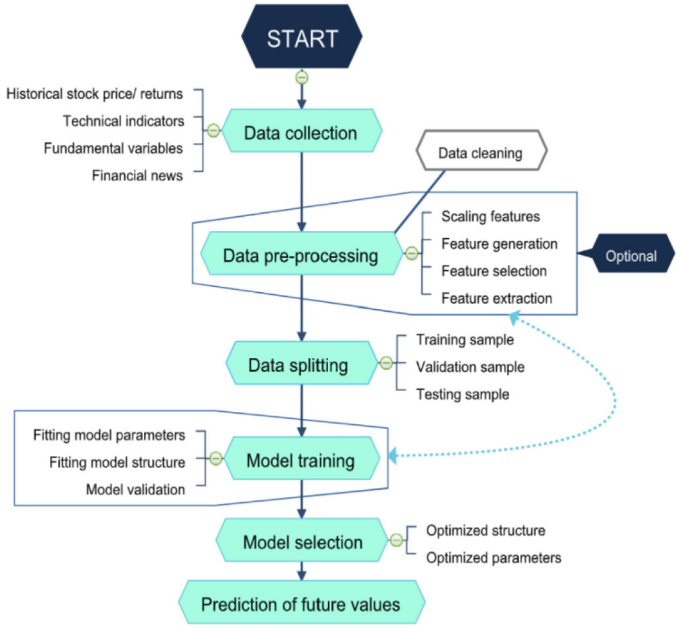

Figure 2: Stock market prediction model workflow with supervised learning [20]

According to Kurani *et al*. [22], SVM is a powerful tool to forecast with little data or real-time analysis required. However, the forecasts are very accurate when applied to a large dataset. With large datasets, SVM requires high computational cost, hence it requires a lot of time to execute the task. Another downfall of SVM is that it is sensitive to the provided data type. So, it is important to normalise the data before training with SVM. The author reviewed and evaluated past recent research using ANN and SVM to forecast stock price. From the analysis, the SVM models are on average 60% to 70% accurate. However, the average accuracy of ANN models was higher than SVM, at 60% to 95%. In a nutshell, ANN outperforms SVM due to its limitations, and it can be concluded that ANN is a better model for forecasting stock price.

The main drawback of the classical neural network is that it is a deterministic neural network, which does not completely represent the stochastic environment of the market. According to Ling *et al*. [23], ANN does not fully represent the variability found in a system's natural circumstances, nor do they convey the complexity of the behaviour of the entire system. The output of the deterministic models is fully determined by the feature's value and initial conditions [24]. The introduction of randomness into these models yields stochastic models. Moreover, stochastic models result in more accurate future value results when the behaviour of the time series is not complex and the initial condition fulfils the assumption of stationary [25]. This review paper focuses on the studies that have integrated stochasticity into ANN or Multilayer Perceptron (MLP).

ANN is based on the biological neural network, which is inspired by the transmission of information in the human brain [23, 26]. ANN is an example of a shallow neural network, and its extension is MLP. MLP is one of the neural networks associated with the Deep Neural Network (DNN). However, in most works, ANN and MLP are terms that are used interchangeably, and in certain studies, MLP is not classified under deep learning. In this article, MLP refers to a neural network with more than one hidden layer whereas ANN refers to a neural network with only one hidden layer.

This paper aims to present a systematic review of hybridised machine learning model between ANN/MLP and the stochastic process. In this article, the research methodology is discussed in the next Research Methods section, and then followed by two sections which review the related manuscripts and the learning theories of the implied neural network, respectively. Discussion on the significance of stochastic neural networks, the limitations and future research direction are presented before concluding the manuscript.

Research Methods

This review paper focuses on MLP/ANN which have been hybridised with the stochastic process. To perform the systematic review, two primary scientific paper databases, Web of Science (WoS) and Scopus, were selected. Titles and topics with "Stochastic Neural Network" are considered to filter out the articles. The filter criteria for the keywords "forecasting" and "prediction" are applied to the category of title, abstract, and keyword. This filtration process is done to narrow the application of SNN to forecasting tasks. Figure 1 shows the flow diagram of the selected articles. From the search 46 and 555 articles were found on Scopus and Web of Science (WoS) databases, respectively. After removing all the duplicate articles, and screening the abstract and full articles, an initial set of seven articles was selected. These articles fulfilled the objective of this review paper, where it should be a hybridisation between MLP and the stochastic process applied on financial forecasting. Another article was included based on intext citation because it was found to be relevant to the research scope. From this survey, it is found that very few cross-disciplinary studies are available on the stochastic process and neural network in the stock market. Hence, this paper provided a comprehensive review of these eight selected articles.

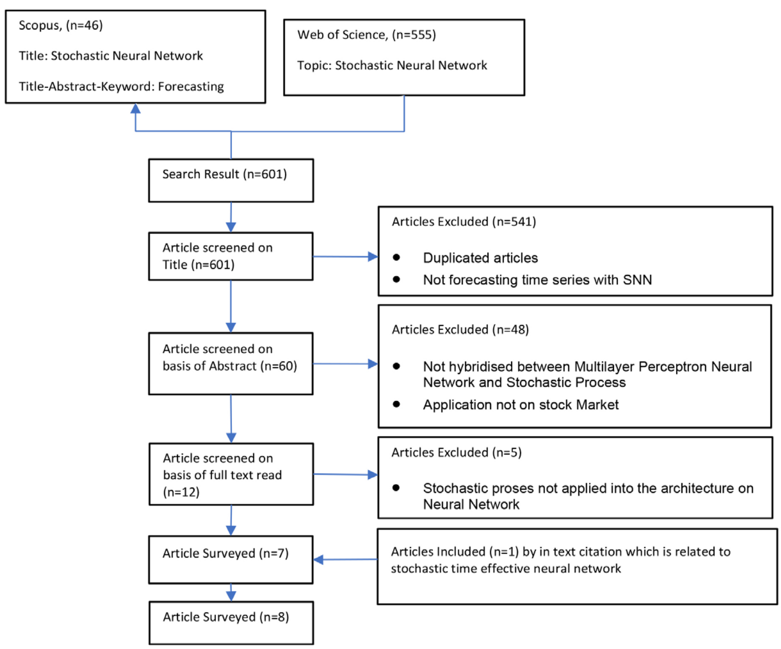

Figure 3: Flow diagram of the selected articles from the database [39]
Note: Abbreviation: n, number of systematic reviews

Literature Review

Stochastic Time Effective Neural Network (STNN) model was first developed by Liao and Wang [26], to forecast the global stock index. The STNN model was developed by incorporating Brownian motion into loss function and applied it into ANN. It is done to allow the model to have the effect of random movement while maintaining the original trend of the financial market. This model was further improved by incorporating the jump process [27], principal component analysis [28], and return scaling cross-correlation function of exponential parameter [29]. Additionally, SNN model was proposed by incorporating random walk theory into the activation function of the neural network to forecast cryptocurrency prices [30]. Table 1 summarises all the selected surveyed articles in this systematic literature review for Figure 1.

The architecture of the proposed STNN and its extension listed in Table 1 had one hidden layer and one output node. However, the number of nodes in the input and hidden layers varied. STNN [26], JTSNN [28, 32] had five input nodes, which comprised daily opening, closing, highest price, lowest price and daily traded volume of the selected stocks. Besides that, four input nodes were considered in the proposed model by Mo and Wang [32], Wang and Wang [33] and for STNN and ANN in the research by Wang and Wang [28]. PCA-STNN model had two input nodes which are obtained through principal component analysis with six variables (daily opening, closing, highest and lowest price, daily traded volume, and daily turnover) [28]. The number of input nodes and hidden layers varied from the ranges of seven to 10 and 11 to 19, respectively for the proposed STSNN model [29]. Furthermore, the number of nodes in hidden layer was 20 in STNN [26]; 13 in JSTNN [28, 32]; eight in STNN [28, 29] and ANN [28]; 9 in PCA-STNN [28]; and the number hidden nodes were selected through validation for EMD-STNN [33]. In summary, these related articles proposed neural networks with input nodes of four or five, and one hidden layer with different number of nodes. Apart from that, the SNN and MLP model consist of 23 input nodes with five hidden layers with each layer containing 130, 100, 50, 25 and 10, respectively.

Moving on to the performance evaluation of these models, average relative error of STNN was less than 10% [26], however the improved JSTNN had a mean relative error less than 5% [27] for all the applied financial datasets. The performance metrics MAE, RMSE, and MAPE of STNN were smaller in comparison to ANN when the next trading day closing price of SSE, SZSE, DJIA, IXIC, and S&P 500 is forecasted [32]. Similarly, the proposed STSNN outperformed the deterministic model (ANN) application of both the China and US stock markets [29]. Empirical results showed that PCA-STNN outperformed the deterministic counterpart, STNN and ANN [28]. Based on the evaluation criteria it was concluded that EMD-STNN had lowest prediction error and highest accuracy than STNN, ANN, and SVM, when NYSE, DAX FTSE, and HIS volatility is forecasted. STNN has the second lowest prediction error followed by ANN and SVM [33]. Jay *et al.* [30] also concluded that SNN outperformed both deterministic modes (MLP and LSTM). From these studies it can be concluded based on the performance metric that neural networks with stochastic models have an advantage over the traditional neural network model to some degree. This is because stochastic models are more accurate in detecting data fluctuations, modelling complex relationships between inputs and outputs, and identifying patterns in data than deterministic ones [29].

Table 1: Summary of selected manuscripts

No.	Authors	Objective	Model	Data	Outcome
1	[26]	To investigate the statistical properties of the fluctuations on the Chinese stock index in comparison with the global stock index.	STNN	SAI – (SSE) SBI – (SZSE) HSI DJI IXIC	From the empirical analysis of the forecasting, it is found that the relative error between SAI and SBI is greater than S&P 500. Thus, it shows that Chinese stock markets fluctuate more than foreign stock markets. From the validity test of the volatility parameter , it is concluded that models with$\sigma(t) = 1$ have the lowest Average Relative Error (ARE). Meanwhile, models with$\sigma(t) = 0$, have no effects of randomisation and $\sigma(t) = 2$, have wild fluctuations. It is concluded that STNN has an accurate prediction after analysing the predicted value with linear regression.
2	[27]	Propose JSTNN to forecast the fluctuation of the time series for the crude oil prices. The relationship between the crude oil market and the stock market is then investigated.	JSTNN	SHCI SZCI SZPI Daqing Shengli SINOPEC	In the presence of small fluctuation, the forecasted value is nearer to the actual value and vice versa when price fluctuation is larger. The results had a positive correlation in both absolute returns of actual data and predicted data when detrended fluctuation analysis is performed.
3	[32]	Investigate and forecast the behaviour of the volatility degrees of returns for the Chinese stock market indexes and some global market indexes.	STNN	SSE SZSE HIS DJIA IXIC S&P500	The relative errors between the forecasted values and actual values are almost 5%. Moreover, STNN outperformed ANN. Different volatility degrees for different threshold values are analysed. It is found out that some prediction results have improved by STNN. But it is also stated that as becomes larger, the forecasting effect declines.
4	[31]	To perform the complex statistical analysis of the simulated data and forecasted data. The effectiveness of the models is then analysed from a macro and micro point of view.	VSS JSTNN	SHCI	It has been concluded that the forecasted value with the JSTNN model is closer to actual data from the micro perspective. Whereas VSS has better performance from the macro point of view.

5	[28]	To develop STNN with PCA for financial time series prediction SSE HS3000 S&P500 DJIA	PCA- STNN	NYSE HS3000 S&P500 DJIA	From the empirical examination of the accuracy of prediction, it is found that the proposed model has better performance than ANN, STNN and PCA-ANN.
6	[29]	To hybridise STNN with a return scaling cross-correlation function of exponential parameter and predict the return scaling cross-correlations between SSE and SZSE.	STSNN	SSE SZSE DJIA IXIC	Through empirical analysis, it is concluded that STSNN has better performance than ANN.
7	[33]	To improve the accuracy of financial time series forecast by hybridising EMD and STNN.	EMD- STNN	SYSE DAX FTSE HSI	Proposed model outperformed STNN, ANN and SVM.
8	[30]	To develop SNN based on random walk theory	MLP LSTM MLP-RW LSTM -RW	BTC ETC LTC	Initially, the proposed model performed slightly poorer than MLP and LSTM. But the performance of the SNN improved after training the perturbation factor via gradient descent backpropagation algorithm.

Learning Theories

First, this section discusses in general the formulation of Multilayer Perceptron (MLP). Then it discusses the integration of stochastic processes as demonstrated in the surveyed articles.

Multilayer Perceptron

MLP is also referred to as Multilayer feed-forward neural network and feed-forward deep network. The pioneer of MLP is ANN, which has only one hidden layer. On the other hand, MLP is composed of three interconnected main layers: An input layer, an arbitrary number of hidden layers, and an output layer. In between each connecting layer, the weight matrix is associated, as shown in Figure 4.

Initially, any small random value nearer to zero is assigned a weight [34]. Then, forward propagation is performed by computing the relationship of two layers from the input to output layer by integrating the activation function or transfer function [35-38]. The general form of the forward propagation is expressed as in Equations 1 to 3. To train the neural netwsork, a backpropagation algorithm is often performed by calculating the error function to modify the interconnected weights and biases.

Furthermore, to forecast multivariate time series data, usually Mean Square Error (MSE), loss function, which is also known as error function, is used as shown in Equation 4.

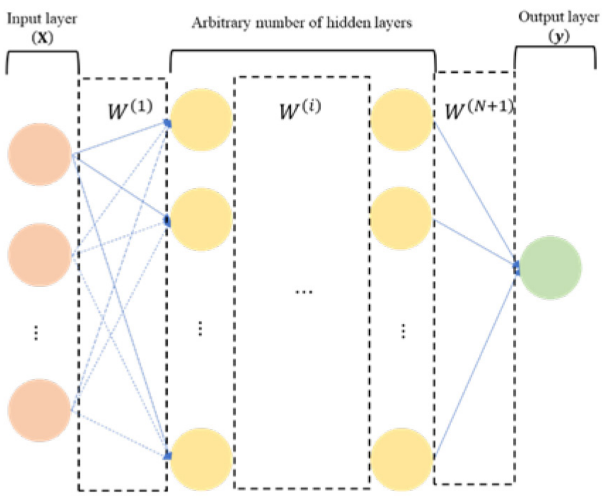

Figure 4: The architecture of MLP with an arbitrary number of hidden layers

$$\boldsymbol{h}^{(1)} = f^{(1)}(W^{(1)T}X + b^{(1)}), \tag{1}$$

$$\boldsymbol{h}^{(i)} = f^{(i)}(W^{(i)T}\boldsymbol{h}(^{(i-1)}b^{(i)}), \tag{2}$$

$$\boldsymbol{y} = f^{(N+1)}(W^{(N+1)T}\boldsymbol{h}(N) + b^{(N+1)}), \tag{3}$$

$$\boldsymbol{E} = \frac{1}{2}(\bar{y} - y)^2$$

where

X	is the input vector,
y	is the output vector,
$W^{(i)}$	is the weight matrix in the layer ,
$h^{(i)}$	denotes the output of hidden layer ,
$f^{(i)}(.)$	denotes the activation function of layer ,
$b^{(i)}$	denotes the bias term in the layer ,
i	and N is the number of last hidden layer,
$\bar{y}$	is the actual value

Stochastic Neural Network Approach

A series of studies on STNN as listed in Table 1, have integrated stochasticity into the error function of MLP. Brownian Motion (BM), a stochastic process, is employed in the studies so that the model can have random movement while maintaining the original trend. The integration of BM into the error function E of Equation 4 is illustrated in Equations 5 to 9. The proposed error function $\bar{E}$ is used to train the neural network via the backpropagation algorithm.

First, the error function for sample (n=1,2,3,.., $\bar{N}$) can be expressed as,

$$E(n,t) = \frac{1}{2}\sum_{k=1}^{\bar{k}}\left(\bar{y}(n) - y(n)\right)^2. \tag{5}$$

The addition of $\phi(t)$ results in Equation (6),

$$E(n,t) = \frac{1}{2}\phi(t)\sum_{k=1}^{\bar{k}}\left(\bar{y}(n) - y(n)\right)^2, \tag{6}$$

where

$$\phi(t_1 - t_n) = \frac{1}{\tau} exp\, exp\left\{\int_{t_1}^{t_n}(t)dt + \int_{t_1}^{t_n}\sigma(t)dB(t)\right\}. \tag{7}$$

Therefore, the error function with stochastic time effective function is,

$$\bar{E} = \frac{1}{\bar{N}}\sum_{n=1}^{\bar{N}}E(n,t), \tag{8}$$

and hence,

$$\bar{E} = \frac{1}{\bar{N}}\sum_{n=1}^{\bar{N}}\frac{1}{\tau}exp\left\{\int_{t_1}^{t_n}\mu(t)dt + \int_{t_1}^{t_n}\sigma(t)dB(t)\right\}\sum_{k=1}^{\bar{k}}(\bar{y}(n) - y(n))^2, \tag{9}$$

where

k	is the number of output nodes, $\{k: 1,2,3, ..., \bar{k}\}$,
$\phi(t)$	is stochastic time strength function,
$\tau(>0)$	is the time strength coefficient,
t_1	is the time of the newest data in the dataset,
t_n	is an arbitrary time point in the data set,
$\mu(t)$	is the drift function,
$\sigma(t)$	is the volatility function,
$B(t)$	is the standard Brownian Motion.

In contrast to all the listed studies, Jay *et al*. [30], stated randomness could be introduced through two approaches. The first would be by randomly tuning a small degree of the value of the weight, but as the network evolves, the feature detection gets noisy, and it may eventually forget the dependencies. Hence, it is emphasised that the first method is not ideal. Moreover, integration of stochasticity into the activation function is ideal as it can interpret the random changes in features. Thus, the authors added random walk (stochastic process) into the activation function, and the generalised formulation of the stochastic behaviour on a neural network is shown in Equation 10. It represents a random-like walk that considers the pattern of the reaction of the market in progressive time steps.

$$S_t = f_t(\cdot) + \gamma\xi_t \times reaction(f(\cdot), S_{t-1}), \qquad 0 < \gamma < 1, \tag{10}$$

with

$$reaction(f_t(\cdot), S_{t-1}) = f_t(\cdot) - S_{t-1},$$

where

S_t	is the vector of post-stochastic operation,
$f_i(\cdot)$	is the activation values of the time step,
γ	is the perturbation factor which controls the amount of stochasticity,
ξ_t	is the operator that produces a vector of random variables of the same dimension as the activation,
$reaction(\cdot)$	is a general function that determines how the current activations will move with respect to the activation of the previous time step.

To sum up, the integration of the stochastic process does not change the architecture of MLP. Still, it only adds stochasticity into the activation function and error function, which are used to optimise the network.

Significance of Stochasticity in Neural Networks

ANN and MLP are efficient models in handling noisy financial data and capture complex relationships between the input and output mapping. They are widely applied in forecasting stock prices because of their self-learning and anti-jamming properties [27-30, 31-33]. However, they are deterministic in nature though applied to noisy environments. According to Ling *et al*. [23], MLP does not have the ability to capture the complexity of the whole system's behaviour. Additionally, large volatilities in the stock markets frequently contribute to the noise in financial time series, making it challenging to include market data directly into a model without making any assumptions. Because stock prices are inherently chaotic and nonstationary, it is difficult to make reliable forecasts [27-30, 31-33]. The introduction of stochastic elements into a neural network became significant because it adapts to the market noise and has the effect of random movement in the model while maintaining the original trend [27-30, 31-33]. Stochastic models are significant in stock price forecasting because they introduce stochastic elements to neural network models, enabling them to capture complex, time-varying relationships in financial data, adapt to market noise, and incorporate random movements while preserving underlying trends. STNNs address the challenges of modelling noisy and nonstationary stock price data.

Research Gap and Recommendations

From this comprehensive review, it can be summarised that the stochastic process was incorporated into deterministic neural networks so that they can mimic and adapt to the original trend of the financial market and improve the accuracy of forecasting. In these reviewed studies, Brownian motion and jump process were incorporated into the loss function, and random walk theory was applied to the activation function. One of the research gaps in the studies related to STNN is that the model considered only one hidden layer. Hence, it is worth extending this research by increasing the number of hidden layers. This is because neural networks with more than one hidden layer perform better [40]. Furthermore, Jay *et al*. [30], proposed SNN by incorporating random walk into the activation function. To improve the neural network, it recommended that Gaussian process, Brownian motion and Jump process be included into the activation function. Finally, a hybridisation between STNN and SNN is also worth applying to forecast the stock price.

Conclusion

From this survey, it can be deduced that forecasting the stock market using a hybridised neural network with a stochastic process (SNN) has better accuracy in comparison to MLP. This is because the introduction of the stochastic process into the neural network makes the model have the effect of random movement while maintaining the original trend. However, from this survey, it can be deduced that there is plenty of room to improve the existing SNN by experimenting with the influential factor, tuning the perturbation factor, and different stochastic processes. Furthermore, the research on the hybridisation of stochastic processes with neural networks has yet to investigate many areas; therefore, the application on financial time series forecasting is worth exploring to improve the neural network's performance. Unfortunately, at this point in time, there are very few studies available in this field; thus, more research is needed.

Acknowledgement

The authors would like to express their gratitude to the Maju Institute of Educational Development (MIED) for their partial financial assistance in supporting this research.

Conflict of Interest Statement

The authors declare that they have no conflict of interest.

References

[1] Johnson, N. F., Jefferies, P., & Hui, P. M. (2003). *Financial market complexity*. Oxford: Oxford University Press. https://doi.org/10.1093/acprof:oso/9780198526650.002.0003

[2] Giles, C. L., Lawrence, S., & Tsoi, A. C. (2001). Noisy time series prediction using recurrent neural networks and grammatical inference. *Machine Learning*, *44*(1), 161-183. https://doi.org/10.1023/A:1010884214864.

[3] Ji, X., Wang, J., & Yan, Z. (2021). A stock price prediction method based on deep learning technology. *International Journal of Crowd Science*, *5*(1), 55-72. https://doi.org/10.1108/IJCS-05-2020-0012

[4] Kumar, D., Sarangi, P. K., & Verma, R. (2022). A systematic review of stock market prediction using machine learning and statistical techniques. *Materials Today Proceedings*, *49*(8), 3187-3191. https://doi.org/10.1016/j.matpr.2020.11.399

[5] Maruddani, D. A. I., & Trimono. (2018). Modeling stock prices in a portfolio using multidimensional geometric brownian motion. *Journal of Physics: Conference Series*, *1025*(1), 012122. https://doi.org/10.1088/1742-6596/1025/1/012122

[6] Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert Systems with Applications*, *42*(1), 259-268. https://doi.org/10.1016/j.eswa.2014.07.040

[7] Islam, M. R., & Nguyen, N. (2020). Comparison of financial models for stock price prediction. *Journal of Risk Financial Management*, *13*(8), 181. https://doi.org/10.3390/jrfm13080181

[8] Mallikarjuna, M., & Rao, R. P. (2019). Evaluation of forecasting methods from selected stock market returns. *Financial Innovation*, *5*(1), 1-16. https://doi.org/10.1186/s40854-019-0157-x

[9] Setyawati, B. R., Creese, R. C., & Jaraiedi, M. (2003). Neural networks for univariate and multivariate time series forecasting keywords. *IIE Annual Conferences Proceeding* (pp. 1-6).

[10] Petchamé, J., Nebot, À., & Alquézar, R. (2012). Quantitative and qualitative approaches for stock movement prediction. *Frontiers in Artificial Intelligence and Applications*, *248*, 233-24. http://dx.doi.org/10.3233/978-1-61499-139-7-233

[11] Atsalakis, G. S., & Valavanis, K. P. (2009). Surveying stock market forecasting techniques - Part II: Soft computing methods. *Expert Systems with Applications*, *36*(3), 5932-5941, 2009, https://doi.org/10.1016/j.eswa.2008.07.006

[12] Wijaya, Y. B., Kom, S., & Napitupulu, T. A. (2010). Stock price prediction: Comparison of Arima and artificial neural network methods - An Indonesia stock's case. *2010 Second International Conference on Advances in Computing, Control, and Telecommunication Technologies, Jakarta, Indonesia, 2010* (pp. 176-179). https://doi.org/10.1109/ACT.2010.45

[13] Mostafa, M. M. (2010). Forecasting stock exchange movements using neural networks: Empirical evidence from Kuwait. *Expert Systems with Applications*, *37*(9), 6302-6309. https://doi.org/10.1016/j.eswa.2010.02.091

[14] Guresen, E., Kayakutlu, G., & Daim, T. U. (2011). Using artificial neural network models in stock market index prediction. *Expert Systems with Applications*, *38*(8), 10389-10397. https://doi.org/10.1016/j.eswa.2011.02.068

[15] Kumar, D. A., & Murugan, S. (2013). Performance analysis of Indian stock market index using neural network time series model. *2013 International Conference on Pattern Recognition, Informatics and Mobile Engineering, Salem, India, 2013* (pp. 72-78). https://doi.org/10.1109/ICPRIME.2013.6496450

[16] Hyup Roh, T. (2007). Forecasting the volatility of stock price index. *Expert Systems with Applications*, *33*(4), 916-922. https://doi.org/10.1016/J.ESWA.2006.08.001

[17] Adebiyi, A. A., Adewumi, A. O., & Ayo, C. K. (2014). Comparison of ARIMA and artificial neural networks models for stock price prediction. *Journal of Applied Mathematics*, *2014*, 614342. https://doi.org/10.1155/2014/614342

[18] Ashish Gajanan Lahane (2008). *Financial forecasting: Comparison of ARIMA, FFNN and SVR* [Report No. 05329R01]. Accessed by September 24, 2023. [Online] from https://www.yumpu.com/en/document/view/23569116/financial-forecasting-comparison-of-arima-ffnn-and-svr

[19] Hiransha, M., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2018). NSE stock market prediction using deep-learning models. *Procedia Computer Science*, *132*, 1351-1362. https://doi.org/10.1016/j.procs.2018.05.050

[20] Kumbure, M. M., Lohrmann, C., Luukka, P., & Porras, J. (2022). Machine learning techniques and data for stock market forecasting: A literature review. *Expert Systems with Applications*, *197*, 116659. https://doi.org/10.1016/j.eswa.2022.116659

[21] Orimoloye, L. O., Sung, M. C., Ma, T., & Johnson, J. E. V. (2020). Comparing the effectiveness of deep feedforward neural networks and shallow architectures for predicting stock price indices. *Expert Systems with Applications*, *139*, 112828. https://doi.org/10.1016/J.ESWA.2019.112828.

[22] Kurani, A., Doshi, P., Vakharia, A., & Shah, M. (2021). A comprehensive comparative study of Artificial Neural Network (ANN) and Support Vector Machines (SVM) on stock forecasting. *Annals of Data Science*, *10*, 183-208. https://doi.org/10.1007/S40745-021-00344-X

[23] Ling, H., Samarasinghe, S., & Kulasiri, D. (2016). Stochastic neural networks for modelling random processes from observed data. In Shanmuganathan, S., Samarasinghe, S. (Eds.), *Artificial neural network modelling*. Studies in Computational Intelligence (Vol. 628, pp. 83-107). https://doi.org/10.1007/978-3-319-28495-8_5

[24] North Carolina State University (n.d.). *Deterministic vs stochastic models* (PowerPoint slides). Accessed by September 11, 2022. [Online] from https://www4.stat.ncsu.edu/~gross/BIO560 webpage/slides/Jan102013.pdf

[25] Hendikawati, P., Subanar, Abdurakhman, & Tarno (2020). A survey of time series forecasting from stochastic method to soft computing. *Journal of Physics: Conference Series*, *1613*(1), 012019. https://doi.org/10.1088/1742-6596/1613/1/012019

[26] Liao Z., & Wang, J. (2010). Forecasting model of global stock index by stochastic time effective neural network. *Expert Systems with Applications*, *37*(1), 834-841. https://doi.org/10.1016/J.ESWA.2009.05.086

[27] Wang, J., Pan, H., & Liu, F. (2012). Forecasting crude oil price and stock price by jump stochastic time effective neural network model. *Journal of Applied Mathematics*, *2012*, 646475, https://doi.org/10.1155/2012/646475

[28] Wang J., & Wang, J. (2015). Forecasting stock market indexes using principle component analysis and stochastic time effective neural networks. *Neurocomputing*, *156*, 68-78, May 2015, https://doi.org/10.1016/J.NEUCOM.2014.12.084

[29] Mo, H. & Wang, J. (2018). Return scaling cross-correlation forecasting by stochastic time strength neural network in financial market dynamics. *Soft Computing*, *22*(9), 3097-3109. https://doi.org/10.1007/s00500-017-2564-0

[30] Jay, P., Kalariya, V., Parmar, P., Tanwar, S., Kumar, N., & Alazab, M. (2020). Stochastic neural networks for cryptocurrency price prediction. *IEEE Access*, *8*, 82804-82818. https://doi.org/10.1109/ACCESS.2020.2990659

[31] Wang, J., Pan, H., Wang, Y., & Niu, H. (2015). Complex system analysis on voter stochastic system and jump time effective neural network of stock market. *International Journal of Computational Intelligence Systems*, *8*(4), 787-795. https://doi.org/10.1080/18756891.2015.1061397

[32] Mo H., & Wang, J. (2013). Volatility degree forecasting of stock market by stochastic time strength neural network. *Mathematical Problems in Engineering*, *2013*, 436795, https://doi.org/10.1155/2013/436795

[33] Wang, J., & Wang, J. (2017). Forecasting stochastic neural network based on financial empirical mode decomposition. *Neural Networks*, *90*, 8-20. https://doi.org/10.1016/J.NEUNET.2017.03.004

[34] Parker, L. E. (2006). *Notes on multilayer, feedforward neural networks*. (CS494/594: Projects in Machine Learning). Accessed by September 11, 2022. [Online] from http://web.eecs.utk.edu/~leparker/Courses/CS594-spring06/handouts/Neural-net-notes.pdf

[35] Ghorbani, M. A., Deo, R. C., Karimi, V., Kashani, M. H., & Ghorbani, S. (2019). Design and implementation of a hybrid MLP-GSA model with multi-layer perceptron-gravitational search algorithm for monthly lake water level forecasting. *Stochastic Environmental Research and Risk Assessment*, *33*(1), 125-147. https://doi.org/10.1007/s00477-018-1630-1

[36] Widegren, P. (2017). *Deep learning-based forecasting of financial assets*. KTH Royal Institution Technology. Accessed by Sep. 11, 2022. [Online] from https://www.math.kth.se/matstat/seminarier/reports/M-exjobb17/170609b.pdf

[37] Sathyanarayana, S. (2014). A gentle introduction to backpropagation. *Numeric Insight*, *7*, 1-15. Accessed by September 11, 2022. [Online] from https://www.researchgate.net/publication/266396438_A_Gentle_Introduction_to_Backpropagation

[38] Svozil, D., Kvasnička, V., & Pospíchal, J. (1997). Introduction to multi-layer feed-forward neural networks. *Chemometrics and Intelligent Laboratory Systems*, *39*(1), 43-62. https://doi.org/10.1016/S0169-7439(97)00061-0

[39] Bustos, O., & Pomares-Quimbaya, A. (2020). Stock market movement forecast: A Systematic review. *Expert Systems with Applications*, *156*, 113464. https://doi.org/10.1016/j.eswa.2020.113464

[40] Vincent A. M. P., & Salleh, H. (2021). An investigation into the performance of the multilayer perceptron architecture of deep learning in forecasting stock. *Universiti Malaysia Terengganu Journal of Undergraduate Research*, *3*(2), 61-68. https://doi.org/10.46754/umtjur.v3i2.205

APPENDIX

List of abbreviations used in Table 1.

STNN	Stochastic Time Effective Neural Network
SAI	Stock A Index
SBI	Stock B Index
HSI	Hang Seng Index
DJIA	Dow Jones Industrial Average
IXIC	Nasdaq Composite
JSTNN	Jump Stochastic Time Effective Neural Network
SHCI	Shanghai Composite Index
SZCI	Shenzhen Composite Index
SZPI	Shenzhen Petrochemical Index
SINOPEC	China Petroleum & Chemical Corporation
SSE	Shanghai Stock Exchange
SZSE	Shenzhen Stock Exchange
VSS	Voter Stochastic System
PCA	Principle Component Analysis
PCA-STNN	Principle Component Analysis - Stochastic Time Effective Neural Network
STSNN	Stochastic Time Strength Neural Network
EMD	Empirical Mode Decomposition
EMD-STNN	Empirical Mode Decomposition - Stochastic Time Effective Neural Network
MLP	Multilayer Perceptron
LSTM	Long Short-Term Memory
MLP-RW	Multilayer Perceptron with random walk
LSTM-RW	Long Short-Term Memory with random walk
BTC	Bitcoin
ETC	Ethereum
LTC	Litecoin
SNN	Stochastic Neural Network
NYSE	New York Stock Index