A New Hybrid Financial Time Series Prediction Model

Bashar Alhnaity[a,*], Maysam Abbod[a]

[a] *Department of Electronic and Computer Engineering, Brunel University London, UK*

Abstract

Due to financial time series' inherent characteristics such as being non-linear, non-stationary, and noisy, with a high degree of uncertainty and hidden relationships, single artificial intelligence and other conventional techniques have failed to capture its non-stationary property and accurately describe its moving tendency. Therefore, researchers and market participants have paid a great deal of attention to tackling such problems. Thus, various models' architecture and new algorithms have been introduced and developed in the literature to alleviate the influence of noise. However, the noise characteristic can refer to the unavailability of information, which affects the behaviour of financial markets and past and future captured prices. The information which excluded in the prediction models is considered as noise, while the non-stationery characteristics imply that the financial time series distributions fluctuate over time. Therefore, the prediction of stock prices and detecting their noise is considered a very challenging and difficult financial topic. This paper adopts a novel three steps hybrid intelligent prediction model which combines a collection of intelligent modelling techniques and feature extraction algorithm. Empirical results show that the proposed model indeed displays good performance in predicting next closing day stock price.

Keywords: Neural Networks, Support Vector Machine, Genetic Algorithm, Ensemble Empirical Mode Decomposition, Financial Time Series.

*Corresponding author
Email address: Alhnaity.Bashar@gmail.com (Bashar Alhnaity)

1. Introduction

Time series is a sequence of observations which taken at the successive time. Extrapolating the discovered patterns from the historical data series into the future is what time series prediction methods attempt to achieve. The prediction domain of financial time series involves the projection of time series such as stock prices, interest rates, inflation, exchange rates and many other financial instruments into the future based on their historical values. In financial predicting, asset prices considered as one of the most discussed topics. Therefore, it is essential for management, asset pricing, portfolio optimisation and selection, option pricing and risk management to fully understand and predict the asset prices. To investors, whether they are individual or financial institutions, the accuracy of the prediction process is of great importance as the prediction output utilised for investment decision making. However, several factors influence assets price such as economical, political and psychological factors [1], and thus these factors introduce high randomness into the financial time series assets price, which reflects on the prediction process and makes it a difficult task.A financial time series characterised natural complexity. Firstly, the behaviour of the financial time series is nearly like a random walk process and also very much like white noise processes. Therefore, under these conditions, it is implied that the theoretical prediction of a financial time series process is impossible [2]. Secondly, financial time series characterised by features such as non-linearity and high non-stationary. Thirdly, they possess structure instability, which implies that external reasons or events influence the movement of the asset price. These events categorised as economic crises or abrupt changes in government policy [3]. Due to these characteristics and the aforementioned sophisticated features of financial time series, predicting them is considered as one of the most challenging tasks in time series prediction. Over the past two decades, many studies have investigated the predictability of financial time series, or more specifically stock prices. In the earlier studies on the predictability of stock prices, it thought that there were no methods that could predict financial time series using his-

torical data sets. Hence, at that time, only the current price could be used as the best predictor of the future price. However, following Malkiel and Fama [4] study on the efficient market hypothesis, prices of stock are informationally efficient, and it is possible to use historical stock prices to predict future prices. Moreover, in the literature, it has been shown that numerous studies employ different suitable methods that can lead to successful prediction values for financial time series. Generally, in the literature, the utilised prediction techniques can be classified into two categories: 1) statistical models, 2) artificial intelligence models (AI). However, statistical models have failed to capture the non-linear patterns that exist in financial time series data as they based on the assumption that a linear correlation exists in them. Therefore, and in order to tackle this limitation, AI techniques have been utilised to enhance the prediction process of financial time series. Besides, the Wolpert theorem [5] stated that no single AI technique could solve all problems. Thus, various models' architecture and new algorithms have been introduced and developed in the literature to alleviate the influence of noise [6]. However, the noise characteristic can refer to the unavailability of information, which affects the behaviour of financial markets and past and future captured prices. The information which excluded in the prediction models is considered as noise, while the non-stationery characteristics imply that the financial time series distributions fluctuate over time. Therefore, the prediction of stock prices and detecting their noise is considered a very challenging and challenging financial topic. Complexity and non-linearity are the main characteristics of the stock market price; thus, it is considered to be a subtle and difficult system for humans to comprehend. Therefore, computational intelligence techniques have been introduced and extensively used in stock prediction. Several studies have been carried out in this area. The recent studies on stock prediction can be categorised into roughly two types: 1. time series prediction [7] [8]; and 2.trend prediction [9] [10]. In time series prediction, the models are trained to fit the historical prices series of an individual stock index and are used to predict the future prices. However, trend prediction models are trained to obtain the relation between various technical variables

and the (rise and decline) movement of stock prices. Generally, in modelling stock trend prediction, a wide range of data mining algorithms have been introduced including neural networks [11], support vector machines [12], logistic regressions [11], decision trees (DT) [11], and naive Bayes [13].In time series modelling, there are several different approaches. Traditional statistical models are linear in their prediction of future value [14]. However, extensive research has introduced and utilised AI techniques such as Artificial Neural Networks (ANNs), Fuzzy Logic, Genetic Algorithms (GAs) and many other techniques; the results showed the prediction capability of such techniques [15] [16] [17]. Initially, theories of model building categorised stock index prediction models into two categories: The first category based on statistical theories such as GARCH and stochastic volatility (SV) [18]. The second category based on artificial intelligence, such as artificial neural networks (ANNs) [19], and the support vector machine (SVM) [1]. The most utilised approaches in many machine learning prediction algorithms for predicting stock price and stock index values are the Artificial Neural Network (ANN) and Support Vector Regression (SVR). However, learning the patterns is not the same in each algorithm. Artificial neural network (ANN) [20] [21], and other statistical prediction methods [22] attempt to predict stock prices under different circumstances such as market atmosphere or economic conditions. Artificial neural networks are one of the most popular prediction techniques. This technique simulates the same learning process as the human biological neural network by developing models from extremely complex and non-linear formulas. It is employed to perform better prediction and analysis output, and different parameters are used to train the neural network. In data mining, preprocessing steps and feature selection can filter redundant and irrelevant features [9]. The results of feature selection in simpler models are a more natural interpretation, and faster induction and basic knowledge [23]. However, identifying more representative features and improving stock prediction are challenging issues, and many studies have claimed that in stock prediction modelling, verifying the feature selection is the success key to this process [9]. In stock prediction models, the most common and adopted fea-

ture selection algorithms include stepwise regression analysis (SRS), principle component analysis (PCA), decision tree (DT), and information gain [9]. All of these feature selection algorithms have only one ability, which is revealing the underlying correlations/associations; they cannot determine the direct influence of stock features on the stock price. Moreover, other analysis methods have been introduced by researchers to enhance prediction accuracy. Different methods from the signal processing area combined with AI techniques and the results were promising. Wavelet Transformation (WT) was combined with a backpropagation neural network by [24]. In their study, the experiment result indicated that the proposed method outperformed the other used method. A novel hybrid model to predict Shanghai securities index by [25] integrated Empirical Mode Decomposition (EMD) with Support Vector Regression. The result from the proposed method indicated that combining EMD with SVR enhanced the prediction performance. The observation is that AI techniques are suitable for stock market prediction. Experiments demonstrate that soft computing techniques outperform conventional models in most cases. They deliver better results than a prediction system with higher predicting accuracy. However, difficulties arise when defining the structure of the models. The proposed methodology in this paper will try to overcome problems such as model structures, data preprocessing, inability to catch dynamics and non-linearity of time series, and lack of understanding of data while taking advantage of the explicit model specification and variable information of stock prices. In this paper, single approach models were used to predict financial time series; however, the results were not satisfactory. Therefore, a novel three steps hybrid intelligent prediction model is adopted based on Back Propagation Neural Network (BPNN), Recurrent Neural Network (RNN), SupportVector Machine (SVR), tuned using Genetic Algorithm (GA) and features extracted using Ensemble Empirical Mode Decomposition (EEMD).

2. Preliminaries

2.1. Research Data

According to Han et al. [26], a time series data set consists of sequences of numeric values obtained over repeated measurements of time. The values typically measured at equal time intervals (e.g. every minute, hour or day). The main assumption that is considered to be a fundamental point of predicting time series is that a pattern existing in historical observations will continue to exist in the future. In other words, this implies that in order to predict the time series, there must be no structural breaks or regime switching in the time series. In this paper, the financial time series that are used to validate the proposed approach is the daily closing prices of three main world indices. The first data set is the FTSE 100 daily closing price from the London stock market. This data set covers the period from 03/01/1984 to 30/10/2014. Figure1 shows the time series plot. The second dataset is the Nikkei 225 daily index of the closing

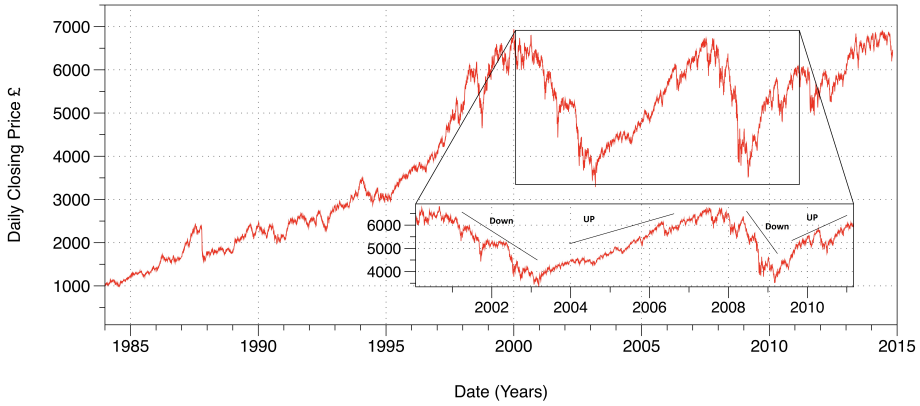

Figure 1: Daily Closing Price for the FTSE100 Index.

price from the Japanese stock market. The data set covers the period from 04/01/1984 to 30/06/2015. Figure2 shows the time series plot. The S&P500 (SPX) index of daily closing price from the USA stock market is the third data set. The data set covers the period from 03/01/1950 to 30/06/2015. Figure 3 shows the time series plot.

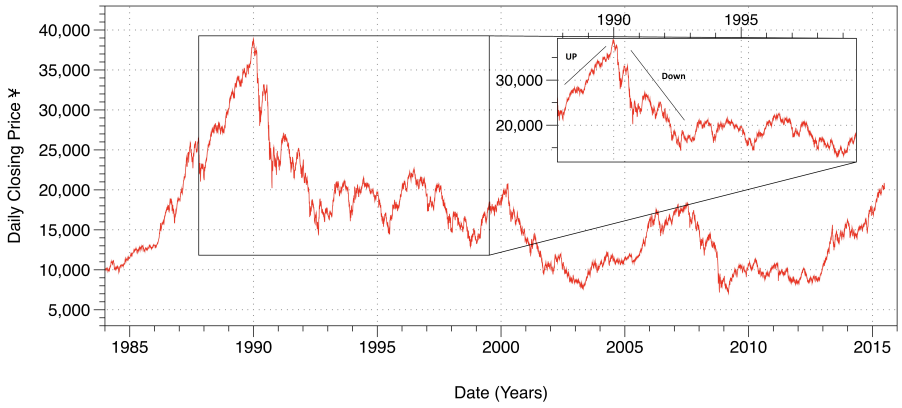

Figure 2: Daily Closing Price for the Nikkie 225 Index.

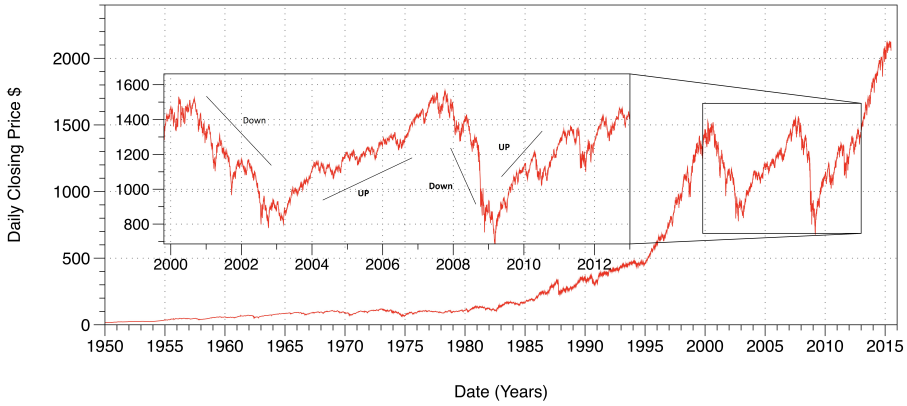

Figure 3: Daily Closing Price for the SPX500 Index.

2.2. In-sample out-of-sample Test

In the given prediction method, in-sample testing is used, utilising all of the available historical data in order to fit a model of interest, which is likely to lead to understating the prediction error, thus overestimating its predictability. The selection and estimation processes within-sample testing are designed to calibrate the prediction procedure to the historical data, and thus, the pattern of these data may not persist into the future. Besides, the selected models by in-sample fit might not be the best to predict out-of-sample data. Therefore, and in order to assess the goodness-of-fit of the proposed method, the predictors

used out-of-sample tests. The evaluation of out-of-sample prediction accuracy begins with the division of the historical data series into a training period (in-sample data) and a test period (out-of-samples data sets). The training period is utilised to identify and estimate a model while the test period reserved for assessing the model's prediction accuracy [27]. The first data set is the daily closing price of the FTSE 100 index, as shown in Subsection 2.1. It is used to demonstrate the predictability of the single approach model. The first 7788 observations (03/01/1984 - 29/10/2013) used as the in-sample data set (training set). The last 250 observations (30/10/2013- 30/10/2014) used as the out-sample set (testing set). The second data set is the daily closing price of the S&P 500 index, as explained in Subsection 2.1, and it is also used to demonstrate and validate the predictability of the proposed single approached method. The first 16230 observations (03/01/1950 - 02/07/2014) used as the in-sample data set (training set). The last 250 observations (03/07/2014 - 30/06/2015) used as the out-sample set (testing set). The third data set is the daily closing price of Nikkie 225 index as Subsection 2.1 illustrated, and it is also utilised to demonstrate the predictability of the proposed single approached methods. The first 7508 observations (04/01/1984 - 04/07/2014) used as the in-sample data set (training set). The last 250 observations (07/07/2014 - 30/06/2015) used as the out-sample set (testing set). Therefore, it is considered a necessary step to divide each data set into two subsets as explained previously, where the in-sample training set is the most extensive, and it is used by the proposed methods to learn the pattern presented in the data. Thus, in order to discover a robust model, it is highly recommended to have a long training duration and large sample [28]. The out-sample testing set, whose length considerably less than the training set, is used to test the performance of the estimated model from the in-sample (training set).

2.3. Ensemble Empirical Mode Decomposition

2.3.1. Empirical Mode Decomposition

Empirical mode decomposition (EMD) is a method that is characterised by the ability to analyse non-linear and non-stationary signals. EEMD performed the EMD over an ensemble of the signal plus Gaussian white noise in order to prevent any mixing problems by populating the whole time- frequency space and benefiting from the dyadic filter bank behaviour of the EMD [29] [30]. The decomposition of the signal defines the EMD transformation mechanism of the signal $x(t)$ into a small number of intrinsic mode functions (IMFs) or modes. There are two conditions the signal must satisfy in order to considered as an IMF: (i) the number of extreme values and the number of zero crossings in the whole data set must either be equal or differ at most by one; (ii) everywhere the mean value of the upper and lower envelope must be zero. As the above point defines the IMFs, the decomposition processes of the signal are explained by the following steps [31]:

- First step: for any signal $x(t)$ the local maxima and minima will be identified. All maxima and maxima will be connected to produce an upper and lower envelop by acubic spline curve. m is defined as the mean value of upper and lower envelopes and the difference between $x(t)$ and m defined as h:

$$h = x(t) - m \tag{1}$$

- Second step: h Will be taken as a new original signal $x(t)$ and then the operation will be repeated in step (a) k times until h is an IMF. To judge whether h is an IMF or not the below function is considered as a termination criterion:

$$D_k = \frac{\sum_{t=0}^{T} \left| h_{(K-1)} h_K(t) \right|^2}{\sum_{t=0}^{T} \left| h_{K-1}(t) \right|^2} \tag{2}$$

  In Equation(2), when D_K is smaller than a predetermined value, h_K can be viewed as an IMF. The first IMF will designate as $c_1 = h_K$.

- Third step: When c_1 is determined, the residue r_1 can be obtained by separating c_1 from the rest of the data as follows:

$$r_1 = x(t) - c_1 \tag{3}$$

After obtaining r_1, the operation will be repeated in steps (1) and (2) as r_1 is the new signal $x(t)$, until obtaining the second $IMFc_2$. Time j will consider until r_j is smaller than a predetermined value or r_j becomes a monotone function in order to obtain all the IMFs. A series of IMFs and a residue r, in the end, will be obtained after applying the above steps. EMD has many drawbacks; mode mixing is one of the main drawbacks which usually cause intermittency in the analysing signal. This drawback can imply either a dingle IMF consisting of signals of dramatically disparate scales or a signal of the same scale appearing in different IMF components. To prevent such problems, a new noise-assisted data analysis method EEMD proposed. In EEMD, the ensemble of the trails' mean is the specific IMF component. A decomposition result of the signal plus a white noise of finite-amplitude consisting of each trail [30]. Several white noise studies have shown that the EMD method is an effective self-adaptive dyadic filter bank when applied to white noise, demonstrating the benefits of the EEMD method [32] [33]. In EEMD, IMF's components are defined as the mean of the corresponding IMFs obtained via EMD over an ensemble of trials, which are generated by adding different realization of white noise of finite variance to the original signal $x[n]$. The below point describes the EEMD algorithm [34]:

- Add white noise series to the original signal. In other words generate $x^i[n] = x[n] + w^i[n]$, where $w^n[n] (i = 1, ..., I)$ are different realization of Gaussian noise.

- Decompose the signal with added white noise into IMFs by EMD. Each $x^i[n] (i = 1, ...I)$, is fully decomposed by EMD getting their modes $IMF_i^K[n]$, where $= 1....K$ indicates the modes.

- Repeating the previous two steps for a certain number of times with differ-

ent white noise each time and obtain the corresponding IMF components of the decomposition.

- Calculate the mean of all the corresponding IMF components and take the mean as the final result for each IMF. In another words assign IMF_K as the $K-th$ mode of $x\left[n\right]$, obtained as the average of the corresponding $IMF_K^i : IMF_K\left[n\right] = \frac{1}{I}\sum_{i=1}^{I} IMF_i^K\left[n\right]$.

2.3.2. Ensemble Empirical Mode Decomposition with Adaptive Noise

The above point explained how, when observing operations in the EMD, each is $x^i\left[n\right]$ decomposed independently from other realizations and so for each one a residue $r_k^i\left[n\right] = r_{k-1}^i\left[n\right] - IMF_k^i\left[n\right]$ is obtained. In this proposed method, the decomposition mode will be noted as $\widetilde{IMF_K}$ and the unique first residue propose as:

$$r_1\left[n\right] = x\left[n\right] - \widetilde{IMF_1}\left[n\right] \tag{4}$$

In Equation (4) $\widetilde{IMF_1}\left[n\right]$ is obtained in the same way as in EEMD. After that, the first EMD is used over an ensemble of r_n plus different realization of a given noise to obtain $\widetilde{IMF_2}$ by averaging. Then, the next residue is defined as: $r_2 = r_1\left[n\right] - \widetilde{IMF_2}\left[n\right]$. Until the stopping criterion is reached, the previous steps with the rest of modes will be continued. This method can be described as following algorithm if $x\left[n\right]$ is the target data.

- To obtain the first modes and compute, decompose by EMD I realization $x\left[n\right] + \varepsilon_0 w^i\left[n\right]$. $\widetilde{IMF_1}\left[n\right] = \frac{1}{i}\sum_{i=1}^{I} IMF_1^i\left[n\right] = \overline{IMF_1}\left[n\right]$.
- In the first stage ($k = 1$)calculate the first residue as in Equation (**??**): $r_1\left[n\right] = x\left[n\right] - \widetilde{IMF_1}\left[n\right]$.
- Decompose realisations $r_1\left[n\right]+\varepsilon_1 E_1(w^i\left[n\right]), i = 1...I$ until their first EMD mode and define the second mode: $\widetilde{IMF_2}\left[n\right] = \frac{1}{I}\sum_{i=1}^{I} E_1\left(r_1\left[n\right] + \varepsilon_1 E_1\left(w^i\left[n\right]\right)\right)$
- For $k = 2...K$ calculate the $k-th$ residue:

$$r_k\left[n\right] = r_{(k-1)}\left[n\right] - \widetilde{IMF_k}\left[n\right] \tag{5}$$

- Decompose realizations $r_k[n] + \varepsilon_k E_k\left(w^i[n]\right), i = 1...I$, Until their first EMD mode and define the $(k+1)-th$ mode as:

$$\widetilde{IMF_{(k+1)}}[n] = \frac{1}{I}\sum_{i=1}^{I} E_1\left(r_k[n] + \varepsilon_k E_k\left(w^i[n]\right)\right) \tag{6}$$

- Then for next k go to step 4. Steps 4 to 6 are performed until it is no longer feasible to be decomposed the obtained residue. The final residue satisfies:

$$R[n] = x[n] - \sum_{k=1}^{K} \widetilde{IMF_k}, \tag{7}$$

When K the total number of modes is complete. Therefore, the given signal $x(n)$ can be expressed as:

$$x[n] = \sum_{k=1}^{K} \widetilde{IMF_k} + R[n]. \tag{8}$$

Equation (8) makes the proposed decomposition complete and provides an exact reconstruction of the original data. At each stage on the observable selection of SNR is allowed by ε_i coefficient. Wu and Hung [**?**] that regarding the amplitude of the added noise, small amplitude values for data dominated should use by high-frequency signals and vice versa.

2.4. Artificial Neural Network

In recent years, predicting financial time series utilising Artificial Neural Networks (ANNs) has increased dramatically. The idea of ANN seen before reaching the output, where the filtration of the inputs through one or more hidden layers, each of which consists of hidden units, or nodes, is considered to be the main idea. Thus, the final output is related to the intermediate output [35]. The ability to learn from data through adaptive changing structure based on external or internal information that flows through the network during the learning phase and generates output variables based on learning is one of the essential advantages of ANN. Furthermore, the non-linear nature of ANN is also

a valuable quality. ANN classified as a non-linear data modelling tool; thus, one of the primary purposes of utilising such a model is to find the patterns in data or to model complex relationships between inputs and outputs. Hence, an explicit model-based approach fails, but ANNs adapts to irregularities and unusual features in a time series. The application of ANNs popularly utilised in financial time series prediction modelling. However, the same as with any other technique, ANNs have some disadvantages such as not allowing much understanding of the data, which might be caused by it not being an explicit model. Therefore, providing a black box for the prediction process is considered as a disadvantage. The danger of over-fitting the in-sample training data is also a major ANN method drawback [36]. In term of the goodness of fit, the performance on in-sample data sets, which ANNs trained on, is reasonably good. However, in out-of-sample sets, its performance is conditional on not breaking the structure in the data sets. According to Balestrassi et al.[37], excessive training time and a large number of parameters that experimentally selected in order to generate useful predictions are considered to be the other drawbacks facing ANN applications. In this paper, two ANN architectures, BPNN and RNN, are used to predict financial data, and subsections 2.4.1 and 2.4.2 demonstrate these models.

2.4.1. Back Propagation Neural Network

In modelling time series with non-linear structures, the most commonly used structure is three layers feed-forward back propagation [38]. The weights determined in the backpropagation process by building connections among the nodes based on data training, producing a least-mean-square error measure of the actual or desired and the estimated values from the output of the neural network. The initial values assigned for the connection weights. In order to update the weights, the error between the predicted and actual output values backpropagated via the network. Minimising the errors in the desired and predicted output attempts take place after the procedure of supervised learning [39]. The architecture of this network contains a hidden layer of neurons

with a non-linear transfer function and an output layer of neurons with a non-linear transfer function and an output layer of neurons with linear transfer functions. Figure 4 illustrates the architecture of a back propagation network,

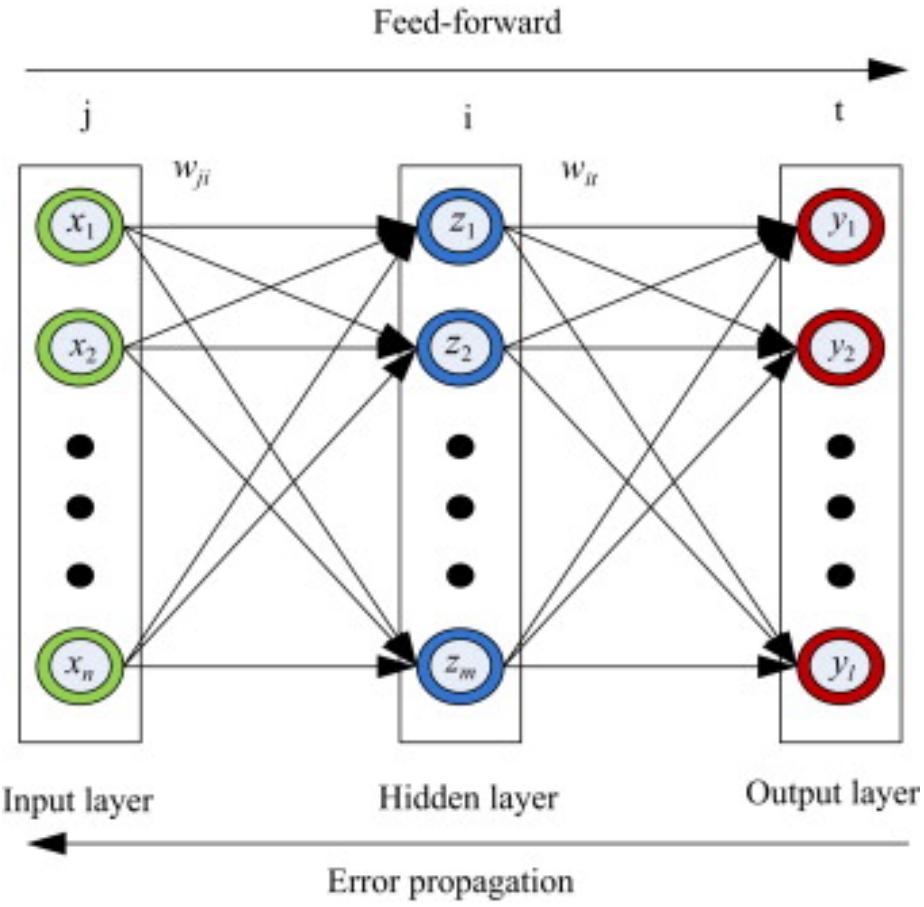

Figure 4: Architecture of feed forward back propagation neural network.

where $x_j\,(j = 1, 2, ..., n)$ represent the input variables; $z_i\,(j = 1, 2, ..., m)$ represent the outputs of neurons in the hidden layer; and $y_t\,(t = 1, 2, ..., l)$ represent the outputs of the neural network [40]. In theory, the neural network has the ability to simulate any kind of data pattern given a sufficient training. Training the neural network will determine the perfect weight to achieve the correct outputs. The following steps illustrate the training process of updating the weights values [41]. The first stage is hidden layers; the below equation explains how

the outputs of all the neurons in the hidden layer are calculated:

$$net_i = \sum_{j=0}^{n} w_{ji} x_j vi = 1, 2, ...m \tag{9}$$

$$z_i = f_H\left(net_i\right) i = 1, 2, ...m \tag{10}$$

where net_i is the activation value of the ith node, z_i is the output of hidden layer, and f_H is called the activation Equation of a node, in this paper a sigmoid function is utilised. Equation 11 explains the utilised sigmoid activation equation.

$$f_H\left(x\right) = \frac{1}{1 + exp(-x)} \tag{11}$$

Second stage the output: The outputs of all the neurons in the output layer are given as equation 12 illustrates:

$$y_t = f_t\left(\sum_{i=0}^{m} w_{it} z_i\right) t = 1, 2, ...l \tag{12}$$

The activation equation is $f_t\left(t = 1, 2, ..., l\right)$, Which is usually a line equation. The weights are assigned with random values initially, and are modified by the delta rule according to the learning samples traditionally. The topology in this study is determined by the trial and error method, which was conducted to choose the best number of neuron experiments different ranges of 20 to 5 neurons in a hidden layer two layers feed-forward backpropagation network tried. The model stopped training after reaching the pre-determined number of epochs. The ideal topology was selected based on the lowest Mean Square Error.

2.4.2. Recurrent Neural Network

Recurrent Neural Network considered as an enhanced ANN architecture, and thus a variant of Elman's network [42]. The ability to form more complex computations than the static feed-forward network is the reason behind adopting the RNN algorithm. Furthermore, the capability of learning temporal pattern

sequences, which are context- or time-dependent is also an advantage of utilising such a method. Embodying a short-term memory by activating a feedback network is also one of the main features of a simple RNN. According to Tenti [43], requiring more substantial memory and connections in simulations, in comparison with a backpropagation network, is one of the main disadvantages of RNN, and therefore this increase leads to high computational timing. However, utilising RNN can yield better results. Figure5 demonstrates the RNN architec-

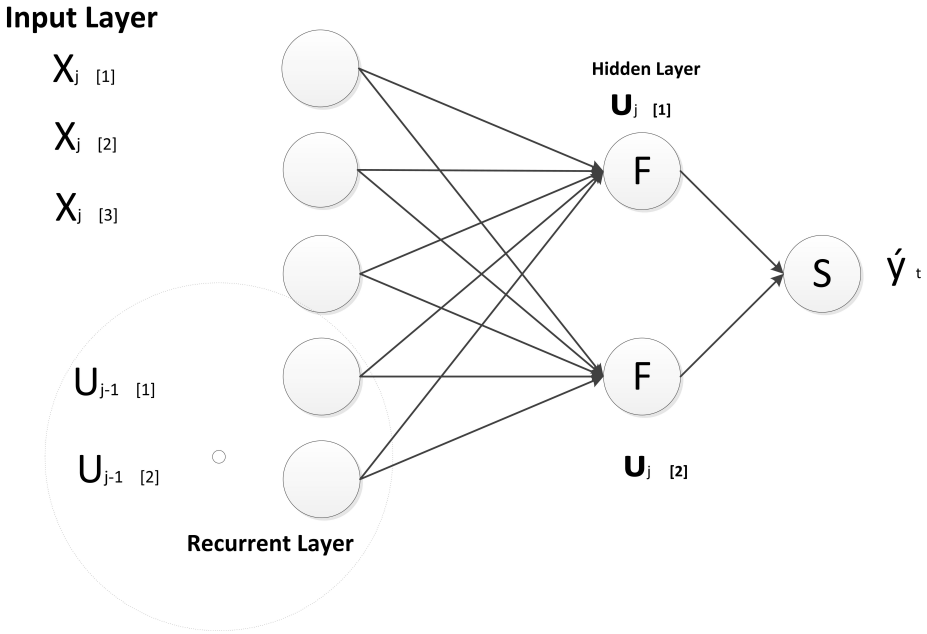

Figure 5: Architecture of Recurrent Neural Network.

ture, where $x_t^{[n]}\,(n=1,2,...k+1)\,, u_t^{[1]}, u_t^{[2]}$ are the inputs for the RNN model at time t including bias node. The output of RNN model is $\check{y}_t$ and $d_t^{[f]}\,(f=1,2)$, $w_t^{[n]}\,(n=1,2,...k+1)$ are presenting the weights of the network. The output of the hidden nodes is $U_t^{[f]}, f=(1,2)$ at time t. The activation function in this model is sigmoid $F:K\,(x)=\frac{1}{1+e^{-x}}$ and S in Figure 5 presents the linear function: $J\,(x)=\sum_i x_i$.Function 13 illustrates the way in which the error is

minimised.

$$E\left(d_t, w_t\right) = \frac{1}{T}\sum_{t=1}^{T}\left(y_t - \hat{y}_t\left(d_t, w_t\right)\right)^2 \tag{13}$$

2.4.3. BPNN and RNN Parameters Determination

Aside from the above settings described, for RNN and BPNN algorithms, there were still several parameters to be decided. The number of the hidden layer, number of neurons (n), the value of learning rate (lr), and momentum constant (mc) are PBNN, RNN models parameters that must be efficiently determined. Since there are no general rules for designing the model topology, a trial and error set was conducted to choose the best number of parameters for both the RNN and PBNN models during the training process. The model stopped training after reaching the predetermined number of epochs. The ideal topology was selected based on the lowest Mean Square Error. According to Chauvin and Rumelhart [44], one hidden layer is sufficient to model a complex system with the desired performance. However, in this paper, two hidden layers are used. The best-observed parameters to set up an RNN model giving the minimum MSR are: First, for the FTSE100 data set model the topology is, [20:5] (20, five nodes for the two hidden layers) with a learning rate of 0.6 and momentum constant of 8. Second, the S&P500 model topology is [20:5] (20, five nodes for the two hidden layers) with a learning rate of 0.4 and momentum constant of 4. Third, for the Nikkei225, it is [20:5] (20, five nodes for the two hidden layers) with a learning rate of 0.3 and the momentum constant of 2. Moreover, also the best-observed parameters to set up a BPNN model are: First, for the FTSE100 data set the model topology is, [20:5] (20, five nodes for the two hidden layers) with a learning rate of 0.3 and momentum constant of 2. Second, the S&P500 model topology is [20:5] (20, five nodes for the two hidden layers) with a learning rate of0.7 and the momentum constant of 10. Third, for Nikkei225, it is [10:5] (10, five nodes for the two hidden layers) with a learning rate of 0.7 and the momentum constant of 10.

2.5. Support Vector Regression

The idea of SVM is to construct a hyperplane or set of hyperplanes in a high or infinite-dimensional space, which used for classification. In the regression problem, the same margin concept used in SVM used. The goal of solving regression problems is to construct a hyperplane that is close to as many of the data points as possible. Choosing a hyperplane with a small norm is considered the main objective, while simultaneously minimising the sum of the distances from the data points to the hyperplane [45]. In the case of solving a regression problem using SVM. SVM became known as the support vector regression (SVR) where the aim is to find a function f with parameters w and b by minimizing the following regression risk:

$$R\left(f\right)=\frac{1}{2}\left(x,w\right)+C\sum_{i=1}^{N}l\left(f\left(x_i\right),y_i\right) \tag{14}$$

where C is a trade-off term, the margin in SVM is the first term which is used in measuring VC- dimension [41].

$$f\left(x,w,b\right)=\left(w,\phi\left(x\right)\right)+b, \tag{15}$$

In the Equation 15 $\phi\left(x\right):x\rightarrow\Omega$ is kernel function, mapping x into in the high dimensional space. SVR and as proposed by [45]. The $-\varepsilon$ insensitive loss function is used as follows:

$$l\left(y,f\left(x\right)\right)=\begin{cases}0,if|y-f\left(x\right)<\varepsilon\\|y-f\left(x\right)|-\varepsilon,Otherwise\end{cases} \tag{16}$$

Equation 17 constrained minimisation problem is equivalent to previous minimisation Equation 14. Min

$$y\left(w,b,\zeta^{*}=\frac{1}{2}\left(w,w\right)+C\sum_{i=1}^{N}\left(\zeta_i+\zeta_{i^{*}}\right)\right) \tag{17}$$

Subject to:

$$y_i - ((w, \phi(x_i) + b)) \leq \varepsilon + \zeta_i, \tag{18}$$

$$((w, \phi(x_i)) + b) - y_i \leq \varepsilon + \zeta_{i^*}, \tag{19}$$

$$\zeta_i^* \geq 0 \tag{20}$$

In sample (x_i, y_i) the ζ_i and ζ_{i^*} measure the up error and down error. Maximizing the dual function or in other words construct the dual problem of this optimization problem (primal problem) by large method is a standard method to solve the above minimization problem. There are four common kernel functions, among all these four kernel function, this study will be utilising radial basis function (RBF). In accordance to [46] RBF kernel function is the most widely applied in SVR. Function 21 is defining the kernel RBF, where the width of the RBF is denoted by σ. Furthermore, Cherkassky and Ma [46] suggested that the value of σ must be between 0.1 and 0.5 in order for SVR model to achieve the best performance. In this paper σ value is determined as 0.1.

$$K(x_i, x_i) = exp\left(\frac{-||x_i - x_j||^2}{2\sigma^2}\right) \tag{21}$$

Two parameters should determine C and γ in the SVR model. However, in the literature, there are no general rules for choosing those parameters. Thus, this paper adopts the most common approach to search the best C and γ values, which is a grid search approach [47]. The grid search approach proposed by Lin et al. [48] using a validation process in order to produce useful generalisations with which to decide parameters. The parameters for the FTSE100 prediction next day closing price is $(C = 100, \gamma = 0.0001)$, which gives the minimum mean square error MSE in the training data set and is the best one for setting up the prediction model. For S&P500 the next day closing price prediction combination parameters are $(C = 100, \gamma = 0.0001)$. Furthermore, the best combination results to set up the SVR prediction next day closing price model of the Nikkei225 parameters are $(C = 100.\gamma = 0.0003)$.

2.5.1. Quantization Factor

This paper adopts a new approach to enhance the prediction output of the SVR model. The quantization factor is for the first time introduced to SVR. In this paper, this factor added to the SVR model. As explained above in the methodology section, SVR the model input is (x_i, y_i). After adding the optimal factors which were determined by trial and error, the optimal factor chosen from a range of factors between 10 and 100. The optimal factor was selected based on the lowest mean square error. The steps below illustrate the change in the input after introducing the quantization factor:

$$Xprim = X_i \div factor \tag{22}$$

$$Yprim = Y_i \div factor \tag{23}$$

The model inputs become $(Xprim_i, Yprim_i)$. After applying the above model (SVR) and to obtain the final prediction result, the chosen factor multiplied with the output of the model as illustrated in Equations 24 and 25:

$$Xprim_{pred} = Xprim \times Qfactor \tag{24}$$

$$Yprim_{pred} = Yprim \times Qfactor \tag{25}$$

This method has been proposed to enhance the performance and prediction ability of SVR techniques. To the best of our knowledge, this is the first time that this approach has been introduced and utilised in financial time series predictions (stocks indices predictions). Furthermore, the tests were carried out on the training set of the three stock indices (FTSE100, S&P500 and Nikkei225). First, for the FTSE100, the best $Qfactor$ in the SVR model is 100. Second, for S&P500, the best $Qfactor$ is 100. Third, for Nikkei225, the best $Qfactor$ is 100.

2.6. Benchmark Prediction Model

In this study a traditional prediction model, the Simple Auto-regressive model (AR) is used, in order to benchmark the performance efficiency of the utilised models. Moreover, the simple average in this paper is used as a benchmark combination method.

2.6.1. Simple Auto-regressive Model

The Auto-regressive (AR) model in this study is used as a benchmark model to evaluate the prediction power between the utilised models based on the relative improvements i root mean square error. Equation 26 illustrates AR model used.

$$y_t = a_1 y\,(t-1) + a_2 y(t-2) + ..., a_t y\,(t-p) \tag{26}$$

In Equation 26 $y\,(t)$ is the predicted stock price based on the past close daily price, $y\,(t) - y\,(t-n)$ and the coefficients of AR model are $a_1 - a_n$. Five lagged daily price is the order of which used in the AR model was varied and found to give better prediction result.

2.6.2. Prediction Combination Techniques

Combining different prediction techniques have been investigated widely in the literature. In short- term predictions, combining the various techniques is more useful, according to [47]. Timmermann [49] stated in his study that using the simple average may work as well as more sophisticated approaches. In this paper, the Simple Average (SA) used as a benchmark combination model. Equation 27 illustrates the calculation of the combination prediction method at time t [50].

$$f_{SM}^t = \left(f_{M1}^t + f_{M2}^t + f_{M3}^t\right) \div 3 \tag{27}$$

3. Prediction Evaluations

3.1. Standard Statistical Measure

Table 1 illustrates the utilised prediction accuracy measure formulas. These statistical analyses, after they computed, will provide the required information

regarding the prediction accuracy and will strengthen the conclusions. All five Mean Square Error, Root Mean Square Error, Mean Absolute Error, Cross-correlation coefficient and Standard Deviation (MSE, RMSE, MAE, R and SD) for the retained statistical error, whereas the lower output is the betterprediction accuracy of the model concerned. However, after considering the pros and cons of the above-mentioned statistical accuracy measures, MSE and RMSE are chosen for performance comparison on the same data set, and for comparison across data sets, MAE utilised in this paper [7].

Table 1: Error metrics equation to measure prediction accuracy

Abbrev.	**Formulas**
MSE	$= \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2$
RMSE	$= \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2}$
MAE	$= \frac{1}{N}\sum_{i=t}^{N}\frac{\lvert x_i - \hat{x_i}\rvert}{\lvert x_i\rvert} =$
R	$= \frac{\sum_{i=1}^{N}(x_i - \bar{x_i})\left(\hat{x_i} - \hat{\bar{x}_i}\right)}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x_i})^2 \sum_{i=1}^{N}\left(\hat{x_i} - \hat{\bar{x}_i}\right)^2}}$
SD	$= \sqrt{\frac{\sum(x_i - \hat{x}_i)^2}{N-1}}$

4. Hybrid modelling and prediction procedures

According to Li and Ma [51], traditional linear methods and the majority of sophisticated non-linear machine learning models have failed to capture the complexity and the non-linearities that exist in financial time series, particularly during periods of uncertainty such the credit crisis in 2008 [52]. Financial time series characteristics imply that the statistical distributions of the time series can change over time. Economic fluctuations may cause the cause of these changes,

or political and environmental events [53] [54]. As a result, it has been verified in the literature that no single method or model works well to capture financial time series characteristics properly and accurately describe its moving tendency, which leads to different and inaccurate financial time series prediction results [55] [56]. In order to address these issues, a novel three steps hybrid model is proposed to alleviate the influence of noise, utilising complete ensemble empirical mode decomposition with adaptive noise, Back Propagation neural network BPNN, Recurrent Neural Network RNN, Support Vector Regression SVR and Genetic Algorithm GA.In the literature there are many methods for combining artificial intelligence techniques with EMD used in the area of stock index prediction [57] [25] [58] [59]. However, to the researcher's knowledge, there are no applications of artificial intelligence techniques combined with EEMD modelling and predicting stock indices. This paper introduces EEMD into SVR, BPNN and RNN in order to enhance the modelling capability, for the first time in the literature. Figure 6 illustrates the overall process of the proposed approach for stock index prediction. The following points illustrate the implementation steps of the new hybrid model: RNN hybrid

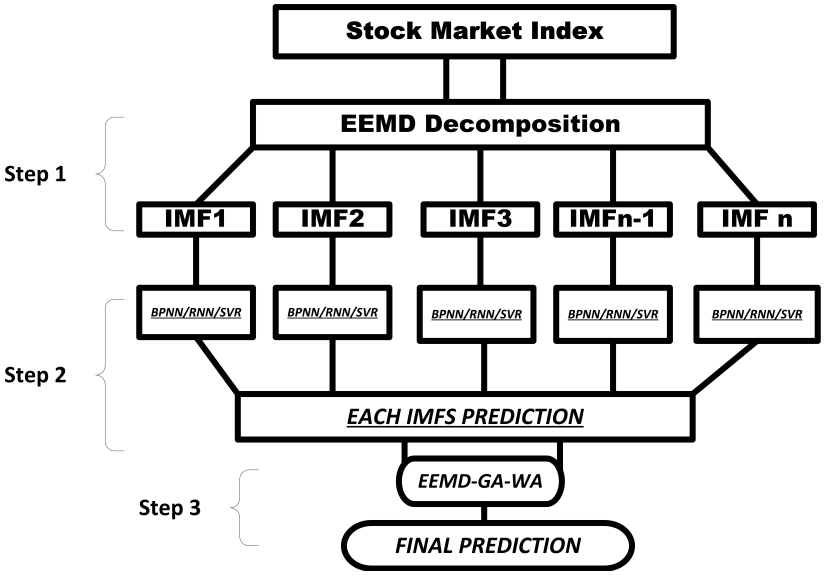

Figure 6: The overall process of EEMD based SVR, BPNN and methodology.

- Step 1: the proposed EEMD has been used to decompose the original stock market index to several intrinsic mode functions. Subsection 2.3 illustrates the data decomposition process by the proposed EEMD technique.

- Step 2: different prediction tools, BPNN, RNN and SVR, have been used as a single predictor to predict the extracted IMFs component and the residue component independently. Besides, the RNN, SVR and RNN methods and model topologies are illustrated (see Subsections2.4 and 2.5). The final prediction value for individual predictor at this step obtained through the sum of the predicted IMFs component and the residue component, Thus, at this stage we obtain three different prediction models, which they are, EEMD-RNN, EEMD-BPNN and EEMD-SVR.

- Step 3: the main goal of this paper is to demonstrate the new hybrid prediction model and examine its ability in the possible improvement of the prediction accuracy of financial time-series data. Therefore, the ensemble weighted average (WA) implemented at this step, where the outputs of a set of separately trained models (EEMD-RNN, EEMD-BPNN and EEMD-SVR) are combined to form one unified prediction. The diversity of the ensemble members achieved using different methods. In this paper EEMD-RNN, EEMD-BPNN and EEMD-SVR used as members, and the ensemble is created using a weighted average, the output of the ensemble is given by a weighted average of its components (WA), as combining method. An optimizer genetic algorithm has been used to determine the weight of the combiner; more details on this step are available in Subsection 4.1 (EEMD-GA-WA). Ensemble simple average (SA), determines the ensemble output by taking the average of all outputs provided by the individual members, is also utilized as a benchmark combination model (EEMD-SA); more details are available in Sub-subsection 2.6.2.

Applying the proposed method can more fully capture the local fluctuations of the original data. Thus, these difficult prediction tasks partially solved by using the decomposition and ensemble principle. Therefore, the proposed method

decomposed the original data into different IMFs where each IMF has simpler frequency components and stronger regularity compared with the original data.Furthermore, EEMD reduced the complexity and improved the efficiency and accuracy of the prediction model. The goal of an ensemble of the ensemble is to formulate a consensus predicting results for the original data. Accordingly, some hybrid ensemble methods motivated by this principle have been proposed recently to solve some tough prediction tasks, and empirical results have demonstrated that hybrid ensemble methodologies outperform individual prediction models in predicting time series data [60] [61] [62] [63] [64].

4.1. Hybrid Combination Model EEMD-GA-weighted factor

Combining different prediction techniques has been investigated widely in the literature. In short- range predictions, combining the various techniques is more useful according to [47], [65], while according to the Timmermann study, using a simple average may work as well as more sophisticated approaches. However, using one model can produce more accurate predictions than any other methods. Therefore, simple averages would not be sufficient in such cases [49]. Compared with different prediction models, the hybrid prediction method is based on a certain linear combination. The assumption for the actual value in period t by model i is $f_{it}\,(i = 1, 2, ...m)$, and the corresponding prediction error will be $e_{it} = y_t - f_{it}$. The weight vector will be $W = [w_1, w_2, ...w_m]^T$ .. Then in the hybrid model the predicted value is computed as follows [66] [67]:

$$\hat{y}_t = \sum_{i=1}^{m} w_i f_{it}\,(t = 1, 2, ...n) \tag{28}$$

$$\sum_{i=1}^{m} w_i = 1 \tag{29}$$

Equation 28 can be expressed in another from:

$$\begin{aligned} &\hat{y} = FW \\ &where \quad \hat{y} = [\hat{y_1}, \hat{y_2}, ...\hat{y_n}]^T, F = [f_{it}]_{n\times m} \end{aligned} \tag{30}$$

The error for the prediction model can be formed as in Equation 31.

$$e_t = y_t - \hat{y}_t = \sum_{i=1}^{m} w_i y_t - \sum_{i=1}^{m} w_i f_{it} = \sum_{i=1}^{m} w_1 w_i \left(y_t - f_{it}\right) = \sum_{i=1}^{m} w_i e_{it} \tag{31}$$

This study proposed a hybrid model that combines EEMD-SVR, EEMD-BPNN and EEMD-RNN.

$$\hat{Y}_{combined_t} = \frac{w_1 \hat{Y}_{M1} + w_2 \hat{Y}_{M2} + w_3 \hat{Y}_{M3}}{w_1 + w_2 + w_3)} \tag{32}$$

(The prediction values in period t are $\hat{Y}_{combined_t}, \hat{Y}_{M1} = \hat{Y}_{EEMD-SVR}, \hat{Y}_{M2} = \hat{Y}_{EEMD-BPNN}$ and $\hat{Y}_{M3} = \hat{Y}_{EEMD-RNN}$ for the hybrid, EEMD-RNN, EEMD-BPNN and EMMD-SVR models, where the assigned weights are w_1, w_2, w_3 respectively, with $\sum_{i=1}^{3} w_i = 1.0 \leq w_i \leq 1$. The most important step in de-

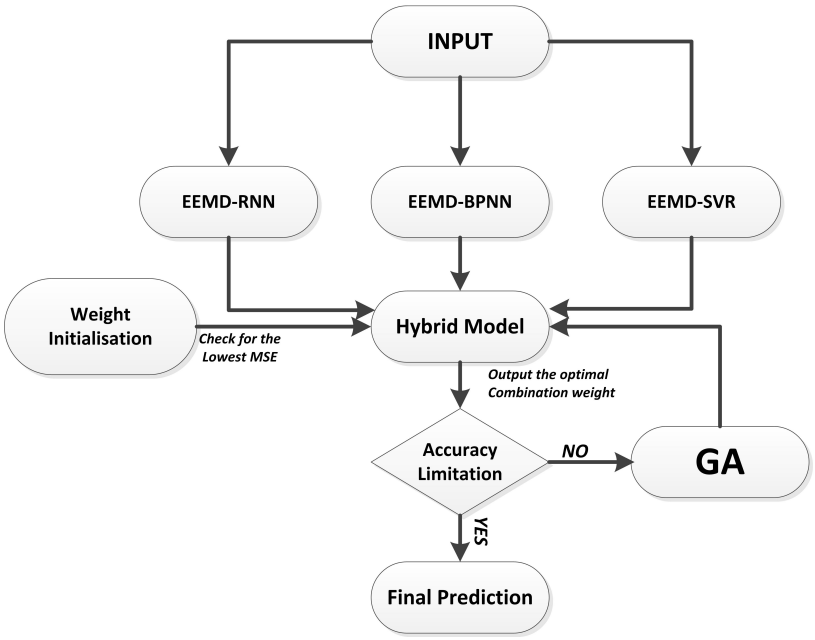

Figure 7: The flow chart of the EEMD-GA-WA hybrid model.

veloping a hybrid prediction model is to determine the perfect weight for each individual model. Setting $w_1 = w_2 = w_3 = 1/3$ in Equation 32 is the simplest

combination method for the three prediction models. Nevertheless, in many cases equal weights cannot achieve the best prediction result. Therefore, this paper adopts a hybrid approach utilising GA as an optimiser to determine the optimal weight for each prediction model. Figure 7 illustrates the architecture of EEMD-GA-WA hybrid model.

4.1.1. Genetic Algorithm

The GA is a popular tool in computational methods modelled on a Darwinian selection mechanism. GA principles were proposed by Holland [68] and developed by Goldberg [69] and Koza [70]. Thus, the main purpose of using such an algorithm is to solve optimisation problems such as determining the optimal weights for the proposed hybrid model in this paper. In comparison with other conventional optimisation methods, GA has many differences, which contribute to making GA more efficient in searching for the optimal solution. The following points exhibit those differences [71].

- Computing the strings in GA algorithms done by encoding and decoding discrete points than using the original parameter values. Thus, GAs tackle problems associated with discontinuity or non-differentiability functions, where traditional calculus methods have failed to work. Therefore, due to the adaptation of the binary strings, such characteristics allow GAs to compute logic operations better.

- The prior information is not essential, and thus, there is no need for such information as the primary population randomly generated. GA uses a fitness function in order to evaluate the suggested solution.

- Initialisation, selection and reproduction, whether crossover or mutation, are what GA depends on in the searching process, which involves random factors. As a result, the searching process in GAs for every single execution will be stand-alone, even the ones under identical parameter settings, which perhaps may affect the results.

5. Experimental Results

In this study, a new hybrid model based on EEMD, SVR, BPNN, RNN and EEMD- GA-WA methods is constructed to predict the FTSE 100, S&P 500 and Nikkei 225 next day closing prices. This section compares the results of the single approach models with the results of the hybrid models. In the suggested model, there are four steps involved in stock index prediction models. Firstly, the original stock index data sets decomposed by EEMD. Secondly, SVR, RNN and BPNN are used to predict each IMF and the residue. Thirdly, in order to obtain the prediction results forthe original data, sets the prediction results of each IMF and the residue must be combined. Fourthly, the predicting results of the three models by the proposed method EEMD-GA-WA combined with the predicting results of the hybrid model. EEMD decomposes the FTSE 100 data set into 13 different IMFs, and IMF 14 is the residue, as shown in Figure 8. In this study, the standard deviation of the noise added is 0.2, and the ensemble size is 500. Figure 8 illustrates the different IMFs with different frequencies. The frequencies from IMF 1 to IMF 6 are much higher, and they mainly reflect the randomness of the information of the original FTSE 100 data set. However, the periodic trends of IMF 7 to IMF 11 are apparent, and they are called the periodic components of the original data set. IMF 12, IMF 13 and IMF 14 are called the trend components. Figure 9 presents the S&P 500 data set after being decomposed by EEMD. As Figure 9 illustrates, different IMFs have different frequencies. From IMF 1 to IMF 9, the frequencies of these IMFs are much higher, and they reflect the randomness of the information of the original S&P 500 data set. IMF 10 to IMF 11 periodic trends are apparent, and these are called periodic components of the original S&P 500 data set. Moreover, IMF 12, 13 and 14 are called the trend components. The decomposed Nikkei 225 by EEMD is shown in Figure 10. As shown in Figure 10 for the decomposed Nikkei 225, different IMFs have different frequencies. Thus, IMF 1 to IMF 8 frequencies are much higher, and they mainly reflect the randomness of the information of the original Nikkei 225 data set. IMF 9 and IMF 10 show the

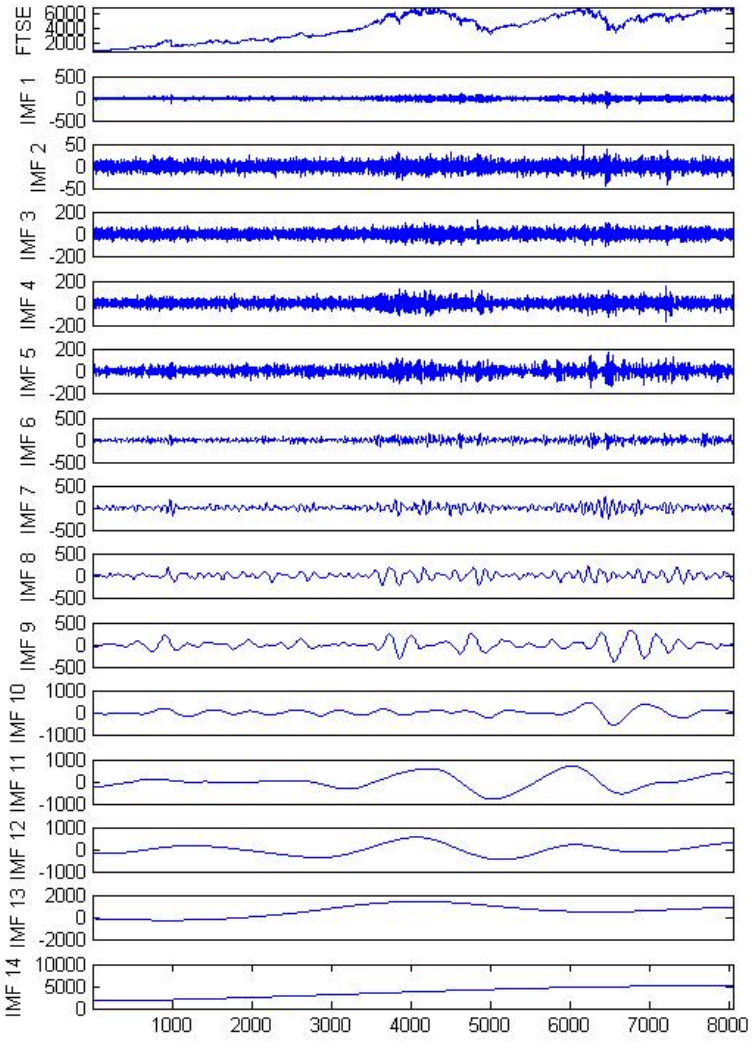

Figure 8: The decomposition of FTSE 100 index closing price

periodic trend, so they are called the periodic components. The rest (IMF 11 to IMF 14) is called the trend components. After FTSE 100, Nikkei 225 and S&P 500 decomposed by EEMD, the prediction problem is changed into predicting

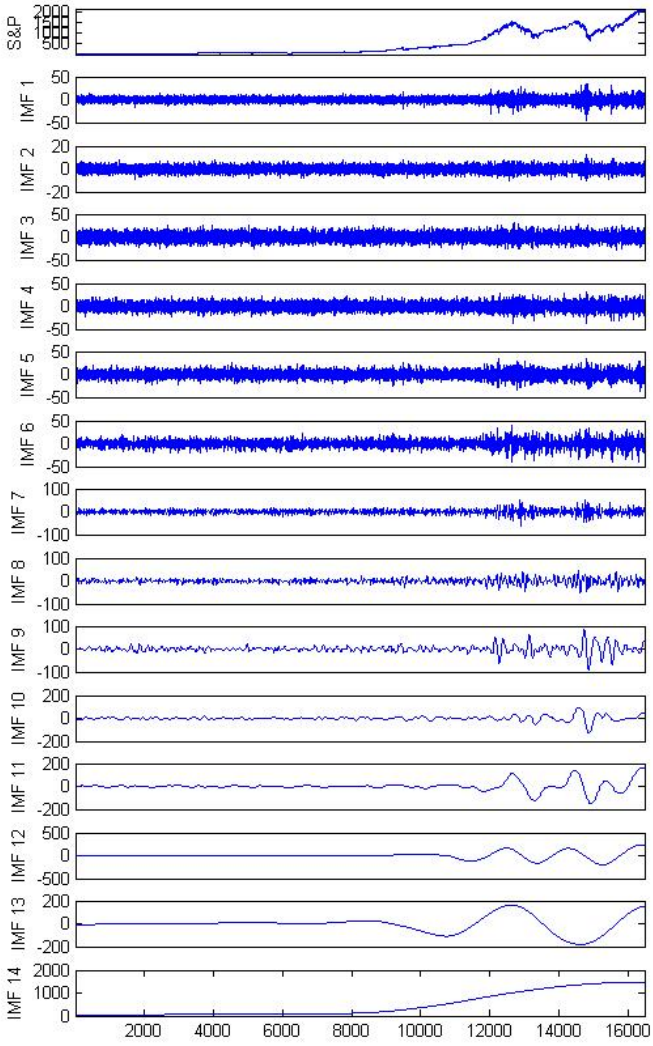

Figure 9: The decomposition of S&P 500 index closing price

each IMF and the residue for the data sets. BPNN, RNN and SVR are used for the predicting of each IMF and the residue. The FTSE 100 decomposed

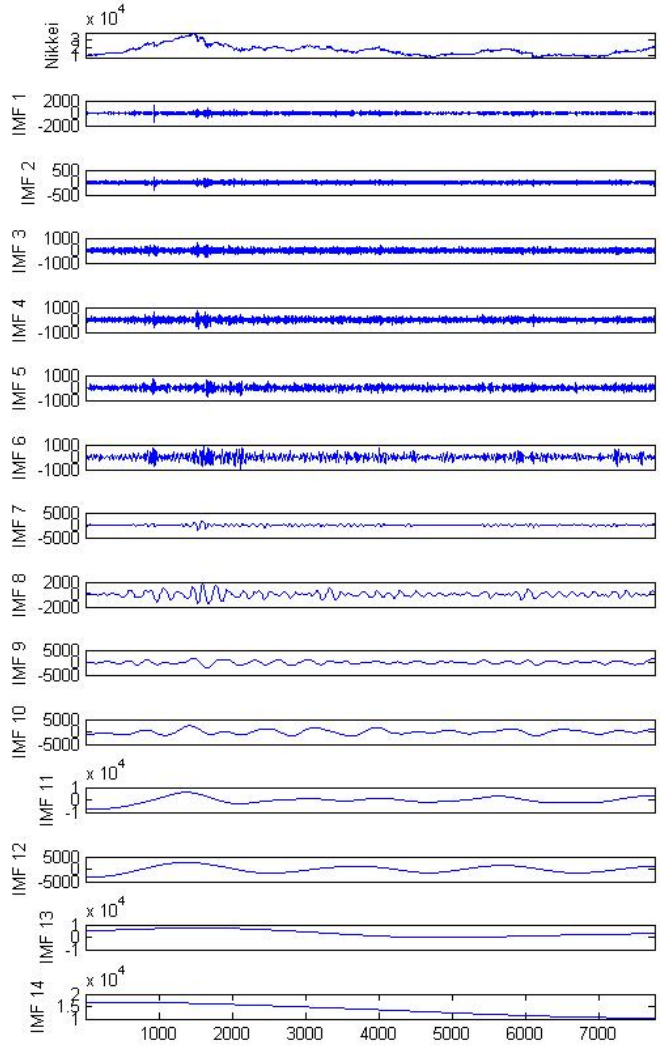

Figure 10: The decomposition of Nikkei 225 index closing price

into 14 IMFs, and in each IMF there are 8038 observations. The first 7788 observations selected as a training data set and the remaining (250) selected

as the testing data set. For S&P 500 there are 14 IMFs, and in each IMF there are 16480 observations. The first 16230 observations set as the training data set andthe remaining (250) observations selected as the testing data set. Finally, for the Nikkei 225, there are 14 IMFs, and in each IMF, there are 7758 observations. The first 7508 selected as the training data set, and the remaining 250 observations selected as the testing data set. Prediction for the FTSE 100, S&P 500 and Nikkei 225 are obtained by merely adding up the prediction results of each IMF and the residue. The simple average (SA) also used in this paper as a benchmark combination method. Besides, the proposed model EEMD-GA-WA prediction results obtained by combining the final results of the three models, as explained in Section 4.1. For the FTSE 100, the predicted results of the methods based on EEMD-RNN, EEMD-BPNN, EEMD-SA and EEMD-GA-WA shown in Figure 11. It observed from Figure 11 that the predicted values obtained from the proposed models EEMD-RNN, EEMD-BPNN, EEMD-SA and EEMD-GA-WA are closer to the actual price than the predicted values of the single models Figure 12. Moreover, the EEMD-GA-WA predicted values are closer to the actual values than with the other models. The actual S&P 500 closing price values and predicted values of the EEMD-SVR, EEMD- RNN, EEMD-BPNN, EEMD-SA and EEMD-GA-WA models presented in Figure 13.

It noticed from Figure 13 that the predicted values from the EEMD-SVR, EEMD- RNN, EEMD-BPNN, EEMD-SA and EEMD-GA-WA models are closer to the actual values of S&P 500 than those from the single approach models from Figure **??**. Moreover, the predicted values from the EEMD-SVR, EEMD-RNN, EEMD-BPNN, EEMD-SA and EEMD-GA-WA models have a smaller deviation between their results and the actual values of the S&P 500 closing prices, as Figure 13 indicates. Figure 15 depicts the actual Nikkei 225 closing price values and predicted values from the EEMD-SVR, EEMD-RNN, EEMD-BPNN, EEMD-SA and EEMD-GA-WA models. From the figure, it observed that the predicted values of all the models have smaller deviations with the actual values. Thus, in comparison with the single-approach- predicted values from Figure 16, the results in Figure 15 indicate that the EMD-SVR, EEMD-

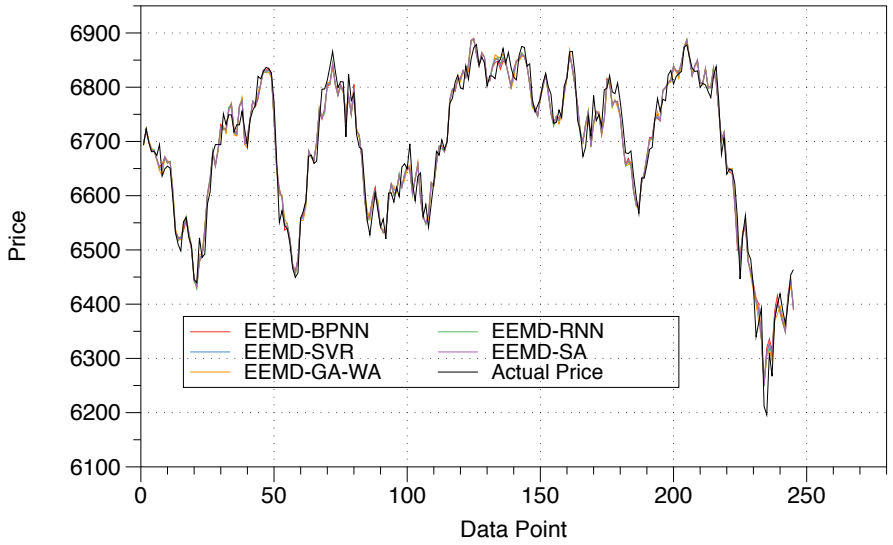

Figure 11: The actual FTSE100 closing price Index and its predicted values from EEMD-SVR, EEMD-RNN, EEMD-BPNN, EEMD-SA and EEMD-GA-WA models.

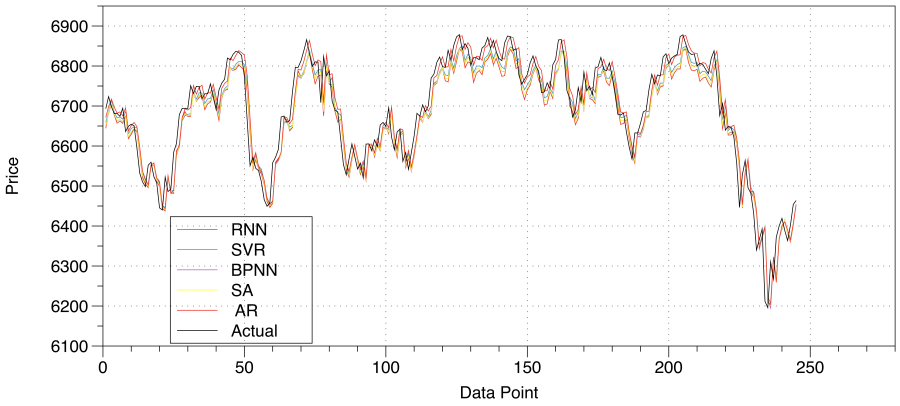

Figure 12: The actual FTSE 100 closing price Index and its predicted values from AR, SVR, RNN, BPNN and SA models.

RNN, EEMD-BPNN, EEMD-SA and EEMD-GA-WA models' predicted values are closer to the actual Nikkei 225 closing prices. The FTSE 100 closing price prediction results of the training data set using EMD- SVR, EEMD- RNN, EEMD-BPNN, EEMD-SA and EEMD-GA-WA models are computed and listed in Table 2. It found from Table 2 that the MSE, RMSE and MAE of the EEMD-

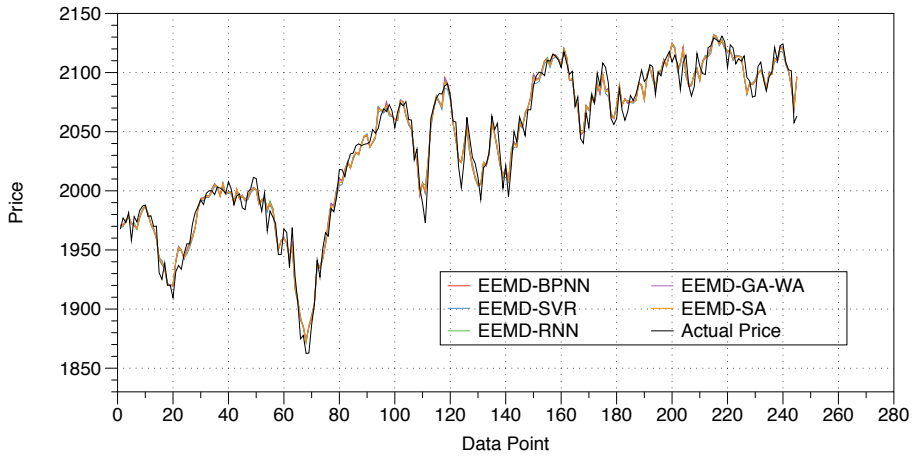

Figure 13: The actual S&P500 closing price Index and its predicted values from EEMD-SVR, EEMD-RNN, EEMD-BPNN, EEMD-SA and EEMD-GA-WA models.

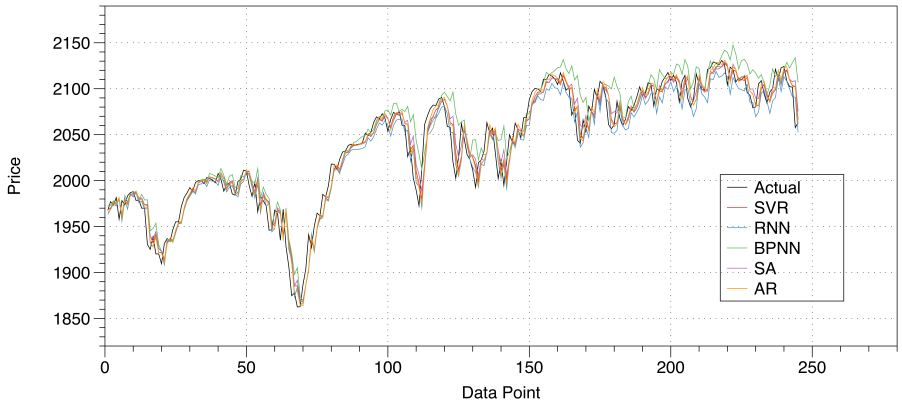

Figure 14: The actual S&P 500 closing price Index and its predicted values from AR, SVR, RNN, BPNN and SA models.

GA-WA model for FTSE 100 predicted values are, respectively, 704.05, 26.53 and 20.15. Thus, it observed that these results are the smallest among the rest of the proposed models. Therefore, it indicates that there is a smaller deviation between the actual and predicted values utilising the suggested model EEMD-GA-WA. Moreover, compared to the predicted results obtained from the single approach in Table 3, the EEMD-GA-WA model has the lowest MSE, RMSE and MAE. The cross-correlation coefficient R results for all prediction models

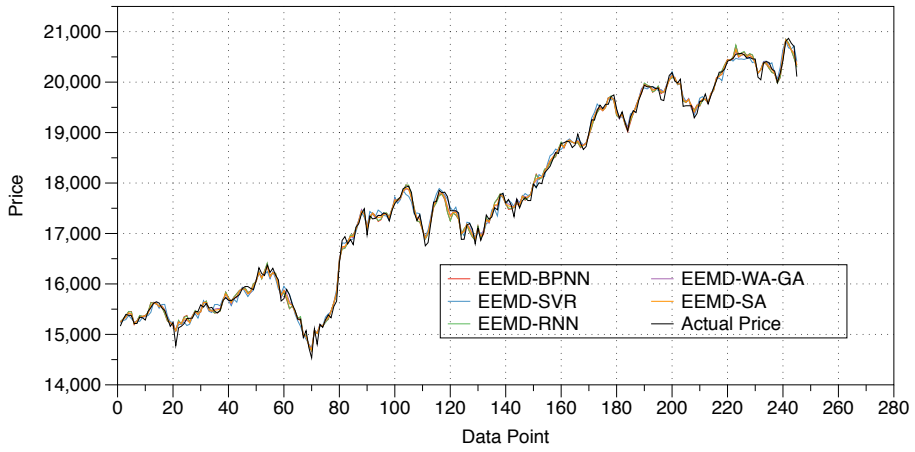

Figure 15: The actual Nikkei 225 closing price Index and its predicted values from EEMD-SVR, EEMD-RNN, EEMD-BPNN, EEMD-SA and EEMD-GA-WA models.

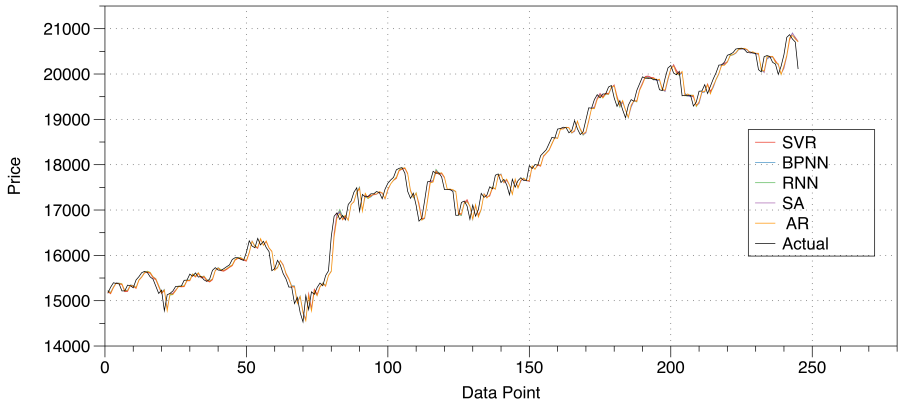

Figure 16: The actual Nikkei 225 closing price Index and its predicted values from AR, SVR, RNN, BPNN and SA.

of FTSE 100 indicate that the prediction values and the actual values do not deviate too much. Besides, each method run twenty times and the standard deviation calculated. Thus, it observed that the results of SD for all models are relatively small, which implies that the models are not running randomly. The predicted values for the training data set of the S&P 500 closing price index illustrated in Table 2. From Table 2, it can be found that the MSE, RMSE and MAE are respectively, 39.40, 6.27 and 2.12 of EEMD-GA-WA. Thus, it

Table 2: The prediction result of training data sets for FTSE 100, S&P 500 and Nikkei 225 using EEMDSVR,EEMDRNN, EEMDBPNN, GA-WA and SA.

Index name	Models	MSE	RMSE	MAE	R	SD
FTSE100	EEMD-SA	714.86	26.73	20.45	0.99	——
	EEMD-SVR	749.70	27.38	20.71	0.99	1.61
	EEMD-RNN	723.54	26.89	20.63	0.99	1.86
	EEMD-BPNN	707.44	26.59	20.41	0.99	1.74
	EEMD-GA-WA	704.05	26.53	20.15	0.99	1.94
S&P500	EEMD-SA	39.11	6.25	3.88	0.99	——
	EEMD-SVR	40.27	6.34	2.58	0.99	1.071
	EEMD-RNN	39.73	6.30	2.61	0.99	1.067
	EEMD-BPNN	39.44	6.28	2.24	0.99	1.19
	EEMD-GA-WA	39.40	6.27	2.18	0.99	1.48
Nikkei225	EEMD-SA	9741.54	98.69	74.47	0.99	——
	EEMD-SVR	12751.14	112.92	85.38	0.99	1.68
	EEMD-RNN	11882.82	109.00	84.14	0.99	1.64
	EEMD-BPNN	12131.96	110.14	84.38	0.99	1.58
	EEMD-GA-WA	9716.02	98.56	74.02	0.99	1.79

Table 3: The prediction result of training data sets for FTSE 100, S&P 500 and Nikkei 225 using SVR,RNN,AR and SA.

Index name	Models	MSE	RMSE	MAE	R	SD
FTSE100	AR	2429.91	49.29	31.93	0.99	——
	SVR	2424.77	49.24	31.89	0.99	2.36
	RNN	2416.33	49.15	32.11	0.99	2.51
	BPNN	2418	49.18	37.65	0.99	2.42
	SA	2417	49.17	37.49	0.92	——
S&P500	AR	58.31	7.63	3.43	0.99	——
	SVR	58.62	7.65	3.45	0.99	2.032
	RNN	57.87	7.60	3.45	0.99	2.021
	BPNN	57.57	7.58	3.46	0.99	2.050
	SA	57.81	7.60	3.45	0.95	——
Nikkei225	AR	55943.34	236.52	160.06	0.99	——
	SVR	53147.7	230.53	157.58	0.99	1.61
	RNN	53676.96	231.67	158.98	0.99	2.32
	BPNN	96473.96	283.66	175.35	0.99	3.64
	SA	67756.20	260	162.92	0.92	——

observed that these results have the smallest error between all utilised models. Therefore, this indicates that there is less deviation between the predicted and actual values utilising the proposed model EEMD-GA-WA. In comparison with the obtained results from the single approach in Table 3, the EEMD-GA-WA results have the lowest error, which implies that the model proposed in this study outperformed the single approach. Table 2 exhibits the cross-correlation coefficient R results for the S&P 500, and it observed that the prediction values and the actual values do not deviate too much. Furthermore, the SD results

Table 4: The prediction result of testing data sets for FTSE 100, S&P 500 and Nikkei 225 using EEMDSVR,EEMDRNN, EEMDBPNN, EEMD-GA-WA and EEMD-SA.

Index name	Models	MSE	RMSE	MAE	R	SD
FTSE100	EEMD-SA	556.31	23.58	17.95	0.98	——
	EEMD-SVR	534.47	23.11	17.58	0.98	1.45
	EEMD-RNN	556.29	23.58	18.05	0.99	1.58
	EEMD-BPNN	598.46	24.46	18.60	0.98	1.67
	EEMD-GA-WA	518.90	22.77	17.53	0.99	1.06
S&P500	EEMD-SA	168.24	12.97	9.03	0.99	——
	EEMD-SVR	170.25	13.04	9.22	0.98	1.38
	EEMD-RNN	175.19	13.23	9.0	0.99	1.54
	EEMD-BPNN	177.63	13.32	9.01	0.99	1.34
	EEMD-GA-WA	118.46	10.88	7.92	0.99	1.86
Nikkei225	EEMD-SA	7895.01	88.85	69.88	0.99	——
	EEMD-SVR	9757.61	98.78	78.63	0.99	1.82
	EEMD-RNN	8632.35	92.91	72.82	0.99	1.53
	EEMD-BPNN	900.5	94	74.91	0.99	1.49
	EEMD-GA-WA	7254.00	85.95	68.47	0.99	1.92

are relatively small, which implies that the models are not running randomly. Table 2 presents the predicted values of the Nikkei 225 closing price for the training data set. As the Table shows, the EEMD-GA-WA model results have the lowest error among all the utilised models, which indicates that there is a smaller deviation between the actual and predicted values using EEMD-GA-WA. Besides, compared to the single approach results in Table 3, EEMD-GA-WA performed much better and achieved the lowest error. Moreover, the results of the cross-correlation coefficient R results for the Nikkei 225, indicate that the deviation between the actual and the predicted values is not that much. Thus, the SD results are relatively small, which implies that the models are not running randomly. The testing data set for the FTSE 100, S&P500 and Nikkei 225 closing price prediction results utilising the proposed methods are computed and listed in Table 4. From Table 4, it found that the MSE, RMSE and MAE of the EEMD-GA-WA model outperformed the rest of the models and have the lowest error, which indicates that there is a smaller deviation between the actual and predicted values utilising EEMD-GA-WA. Moreover, compared to the results of the single approaches in Table 5, the EEMD-GA-WA, EEMD-SVR, EEMD- RNN, EEMD-BPNN and EEMD-SA models have the lowest MSE, RMSE and MAE. Thus, it concluded that introducing EEMD to predict the FTSE 100, Nikkei 225 and S&P 500 has enhanced the obtained

Table 5: The prediction result of testing data sets for FTSE 100, S&P 500 and Nikkei 225 using SVR,RNN,AR and SA.

Index name	Models	MSE	RMSE	MAE	R	SD
FTSE100	AR	1815.46	42.60	31.95	0.95	——
	SVR	1912.04	43.72	34.43	0.95	1.48
	RNN	2248.38	47.23	38.10	0.95	2.34
	BPNN	2188	46.75	37.65	0.95	1.44
	SA	2033.77	45.09	35.99	0.95	——
S&P500	AR	240.56	15.51	11.68	0.96	——
	SVR	244.99	15.65	11.83	0.97	1.21
	RNN	336.28	18.33	14.42	0.96	2.05
	BPNN	602.97	22.85	18.12	0.94	3.45
	SA	267.22	16.34	12.21	0.96	——
Nikkei225	AR	34806.38	186.56	134.88	0.99	——
	SVR	34720.67	186.33	135.36	0.99	2.92
	RNN	34822.62	186.60	135.39	0.99	2.70
	BPNN	35547.91	188.50	137.028	0.99	4.00
	SA	34761.99	186.44	135.32	0.99	——

results and provided a better predict result than the single approach models in terms of prediction error and accuracy.

6. Conclusion

This study proposed a new three-step prediction model by integrating EEMD and RNN, BPNN, SVR, GA-WA for financial time series. The proposed models (EEMD- RNN, EEMD-SVR, EEMD- PBNN, EEMD-SA and EEMD-GA-WA) first use EEMD to decompose the FTSE 100, Nikkei 225 and S&P 500 into different IMFs. Since the financial time series data is inherently noisy, EEMD is utilised to reconstruct the criterion of the original unsteady and non-linear data into specific components which have fixed frequency and periodicity. Furthermore, SVR, PBNN and RNN applied to the predictions of each IMF. After each IMF predicted in the built RNN, PBNN and SVR models, the final prediction results obtained by adding the predicted results of each IMF for each model. One of the contributions of this paper is introducing EEMD as a preprocessor to add white noise and decompose the raw data into a finite set of IMFs, which have high correlations and a more natural frequency. Moreover, this study compared the obtained results of the proposed model with the single

approach models. Thus, the empirical results showed that using EEMD as a preprocessor advances the simplification of RNN, SVR and BPNN modelling and also obtains a much more precise model than the single approach models. Therefore, the new proposed model proved that it is better at predicting non-linear and robust noise data than any other single model employed in this paper. Besides, the weighted average combination method using GA to optimise the weight (the EEMD- GA-WA model) was introduced to combine the final prediction results of EEMD-SVR, EEMD- PBNN and EEMD-RNN in order to enhance the prediction accuracy and minimise the error. According to the experiments, the proposed hybrid model EEMD-GA-WA produced lower prediction errors than the EEMD-SVR, EEMD-PBNN and EEMD-RNN models. Therefore, it concluded that the proposed hybrid model EEMD-GA-WA outperformed the EEMD-SVR, EEMD- PBNN and EEMD-RNN models.

References

[1] W. Huang, Y. Nakamori, S.-Y. Wang, Forecasting stock market movement direction with support vector machine, Computers & Operations Research 32 (10) (2005) 2513–2522.

[2] T. Hellström, K. Holmström, Predictable patterns in stock returns, in: University, Department of, Citeseer, 1998.

[3] H. Schaller, S. V. Norden, Regime switching in stock market returns, Applied Financial Economics 7 (2) (1997) 177–191.

[4] E. F. Fama, Efficient capital markets: A review of theory and empirical work*, The journal of Finance 25 (2) (1970) 383–417.

[5] D. H. Wolpert, W. G. Macready, No free lunch theorems for optimization, Evolutionary Computation, IEEE Transactions on 1 (1) (1997) 67–82.

[6] L.-Y. Wei, A hybrid anfis model based on empirical mode decomposition for stock time series forecasting, Applied Soft Computing 42 (2016) 368–376.

[7] Q. Cao, K. B. Leggio, M. J. Schniederjans, A comparison between fama and french's model and artificial neural networks in predicting the chinese stock market, Computers & Operations Research 32 (10) (2005) 2499–2512.

[8] M. R. Hassan, K. Ramamohanarao, J. Kamruzzaman, M. Rahman, M. M. Hossain, A hmm-based adaptive fuzzy inference system for stock market forecasting, Neurocomputing 104 (2013) 10–25.

[9] C.-F. Tsai, Y.-C. Hsiao, Combining multiple feature selection methods for stock prediction: Union, intersection, and multi-intersection approaches, Decision Support Systems 50 (1) (2010) 258–269.

[10] C.-H. Cheng, L.-Y. Wei, Y.-S. Chen, Fusion anfis models based on multi-stock volatility causality for taiex forecasting, Neurocomputing 72 (16) (2009) 3462–3468.

[11] C.-F. Tsai, Y.-C. Lin, D. C. Yen, Y.-M. Chen, Predicting stock returns by classifier ensembles, Applied Soft Computing 11 (2) (2011) 2452–2459.

[12] M.-C. Lee, Using support vector machine with a hybrid feature selection method to the stock trend prediction, Expert Systems with Applications 36 (8) (2009) 10896–10904.

[13] Y. Zuo, E. Kita, Stock price forecast using bayesian network, Expert Systems with Applications 39 (8) (2012) 6729–6737.

[14] T. S. Rao, M. Gabr, An introduction to bispectral analysis and bilinear time series models, Vol. 24, Springer Science & Business Media, 2012.

[15] E. Hadavandi, H. Shavandi, A. Ghanbari, Integration of genetic fuzzy systems and artificial neural networks for stock price forecasting, Knowledge-Based Systems 23 (8) (2010) 800–808.

[16] Y.-S. Lee, L.-I. Tong, Forecasting time series using a methodology based on autoregressive integrated moving average and genetic programming, Knowledge-Based Systems 24 (1) (2011) 66–72.

[17] M. Zarandi, E. Hadavandi, I. Turksen, A hybrid fuzzy intelligent agent-based system for stock price prediction, International Journal of Intelligent Systems 27 (11) (2012) 947–969.

[18] G. B. Durham, Sv mixture models with application to s&p 500 index returns, Journal of Financial Economics 85 (3) (2007) 822–856.

[19] C. Chen, Neural networks for financial market prediction, in: Neural Networks, 1994. IEEE World Congress on Computational Intelligence., 1994 IEEE International Conference on, Vol. 2, IEEE, 1994, pp. 1199–1202.

[20] R. Pino, J. Parreno, A. Gomez, P. Priore, Forecasting next-day price of electricity in the spanish energy market using artificial neural networks, Engineering Applications of Artificial Intelligence 21 (1) (2008) 53–62.

[21] X. Zhu, H. Wang, L. Xu, H. Li, Predicting stock index increments by neural networks: The role of trading volume under different horizons, Expert Systems with Applications 34 (4) (2008) 3043–3054.

[22] F. Chen, C. Gou, X. Guo, J. Gao, Prediction of stock markets by the evolutionary mix-game model, Physica A: Statistical Mechanics and its Applications 387 (14) (2008) 3594–3604.

[23] Y.-S. Chen, C.-H. Cheng, Evaluating industry performance using extracted rgr rules based on feature selection and rough sets classifier, Expert Systems with Applications 36 (5) (2009) 9448–9456.

[24] T.-J. Hsieh, H.-F. Hsiao, W.-C. Yeh, Forecasting stock markets using wavelet transforms and recurrent neural networks: An integrated system based on artificial bee colony algorithm, Applied soft computing 11 (2) (2011) 2510–2525.

[25] W. Wang, H. Zhao, Q. Li, Z. Liu, A novel hybrid intelligent model for financial time series forecasting and its application, in: Business Intelligence and Financial Engineering, 2009. BIFE'09. International Conference on, IEEE, 2009, pp. 279–282.

[26] J. Han, M. Kamber, J. Pei, Data mining: concepts and techniques: concepts and techniques, Elsevier, 2011.

[27] L. J. Tashman, Out-of-sample tests of forecasting accuracy: an analysis and review, International Journal of Forecasting 16 (4) (2000) 437–450.

[28] D. Zhang, L. Zhou, Discovering golden nuggets: data mining in financial application, Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 34 (4) (2004) 513–522.

[29] T. Wang, M. Zhang, Q. Yu, H. Zhang, Comparing the applications of emd and eemd on time–frequency analysis of seismic signal, Journal of Applied Geophysics 83 (2012) 29–34.

[30] Z. Wu, N. E. Huang, Ensemble empirical mode decomposition: a noise-assisted data analysis method, Advances in adaptive data analysis 1 (01) (2009) 1–41.

[31] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, H. H. Liu, The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis, in: Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, Vol. 454, The Royal Society, 1998, pp. 903–995.

[32] P. Flandrin, G. Rilling, P. Goncalves, Empirical mode decomposition as a filter bank, Signal Processing Letters, IEEE 11 (2) (2004) 112–114.

[33] Z. Wu, N. E. Huang, A study of the characteristics of white noise using the empirical mode decomposition method, in: Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, Vol. 460, The Royal Society, 2004, pp. 1597–1611.

[34] M. E. Torres, M. Colominas, G. Schlotthauer, P. Flandrin, et al., A complete ensemble empirical mode decomposition with adaptive noise, in: Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on, IEEE, 2011, pp. 4144–4147.

[35] J. G. De Gooijer, R. J. Hyndman, 25 years of time series forecasting, International journal of forecasting 22 (3) (2006) 443–473.

[36] B. LeBaron, A. S. Weigend, A bootstrap evaluation of the effect of data splitting on financial time series, Neural Networks, IEEE Transactions on 9 (1) (1998) 213–220.

[37] P. P. Balestrassi, E. Popova, A. d. Paiva, J. M. Lima, Design of experiments on neural network's training for nonlinear time series forecasting, Neurocomputing 72 (4) (2009) 1160–1178.

[38] L. A. Diaz-Robles, J. C. Ortega, J. S. Fu, G. D. Reed, J. C. Chow, J. G. Watson, J. A. Moncada-Herrera, A hybrid arima and artificial neural networks model to forecast particulate matter in urban areas: the case of temuco, chile, Atmospheric Environment 42 (35) (2008) 8331–8340.

[39] M. Kubat, Neural networks: a comprehensive foundation by simon haykin, macmillan, 1994, isbn 0-02-352781-7. (1999).

[40] J.-J. Wang, J.-Z. Wang, Z.-G. Zhang, S.-P. Guo, Stock index forecasting based on a hybrid model, Omega 40 (6) (2012) 758–766.

[41] D. Ö. Faruk, A hybrid neural network and arima model for water quality time series prediction, Engineering Applications of Artificial Intelligence 23 (4) (2010) 586–594.

[42] J. L. Elman, Finding structure in time, Cognitive science 14 (2) (1990) 179–211.

[43] P. Tenti, Forecasting foreign exchange rates using recurrent neural networks, Applied Artificial Intelligence 10 (6) (1996) 567–582.

[44] Y. Chauvin, D. E. Rumelhart, Backpropagation: theory, architectures, and applications, Psychology Press, 1995.

[45] Y. Xia, Y. Liu, Z. Chen, Support vector regression for prediction of stock trend, in: Information Management, Innovation Management and Industrial Engineering (ICIII), 2013 6th International Conference on, Vol. 2, IEEE, 2013, pp. 123–126.

[46] V. Cherkassky, Y. Ma, Practical selection of svm parameters and noise estimation for svm regression, Neural networks 17 (1) (2004) 113–126.

[47] Y. Zhang, L. Wu, Stock market prediction of s&p 500 via combination of improved bco approach and bp neural network, Expert systems with applications 36 (5) (2009) 8849–8854.

[48] C.-W. Hsu, C.-C. Chang, C.-J. Lin, et al., A practical guide to support vector classification (2003).

[49] A. Timmermann, Forecast combinations, Handbook of economic forecasting 1 (2006) 135–196.

[50] C. Xiang, W. Fu, Predicting the stock market using multiple models, in: Control, Automation, Robotics and Vision, 2006. ICARCV'06. 9th International Conference on, IEEE, 2006, pp. 1–6.

[51] Y. Li, W. Ma, Applications of artificial neural networks in financial economics: a survey, in: Computational Intelligence and Design (ISCID), 2010 International Symposium on, Vol. 1, IEEE, 2010, pp. 211–214.

[52] K. Theofilatos, A. Karathanasopoulos, P. Middleton, E. Georgopoulos, S. Likothanassis, Modeling and trading ftse100 index using a novel sliding window approach which combines adaptive differential evolution and support vector regression, in: Artificial Intelligence Applications and Innovations, Springer, 2013, pp. 486–496.

[53] J. W. Hall, Adaptive selection of us stocks with neural nets, Trading on the edge: neural, genetic, and fuzzy systems for chaotic financial markets. New York: Wiley (1994) 45–65.

[54] Y. S. Abu-Mostafa, A. F. Atiya, Introduction to financial forecasting, Applied Intelligence 6 (3) (1996) 205–213.

[55] G. P. Zhang, Time series forecasting using a hybrid arima and neural network model, Neurocomputing 50 (2003) 159–175.

[56] C. Chatfield, What is the bestmethod of forecasting?, Journal of Applied Statistics 15 (1) (1988) 19–38.

[57] H. Yu, H. Liu, Improved stock market prediction by combining support vector machine and empirical mode decomposition, in: Computational Intelligence and Design (ISCID), 2012 Fifth International Symposium on, Vol. 1, IEEE, 2012, pp. 531–534.

[58] C.-H. Cheng, L.-Y. Wei, A novel time-series model based on empirical mode decomposition for forecasting taiex, Economic Modelling 36 (2014) 136–141.

[59] K. Guhathakurta, I. Mukherjee, A. R. Chowdhury, Empirical mode decomposition analysis of two different financial time series and their comparison, Chaos, Solitons & Fractals 37 (4) (2008) 1214–1227.

[60] F. A. Anifowose, J. Labadin, A. Abdulraheem, Hybrid intelligent systems in petroleum reservoir characterization and modeling: the journey so far and the challenges ahead, Journal of Petroleum Exploration and Production Technology 7 (1) (2017) 251–263.

[61] F. Anifowose, J. Labadin, A. Abdulraheem, Improving the prediction of petroleum reservoir characterization with a stacked generalization ensemble model of support vector machines, Applied Soft Computing 26 (2015) 483–496.

[62] R. Ž. Jovanović, A. A. Sretenović, B. D. Živković, Ensemble of various neural networks for prediction of heating energy consumption, Energy and Buildings 94 (2015) 189–199.

[63] A. Onan, S. Korukoğlu, H. Bulut, A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification, Expert Systems with Applications 62 (2016) 1–16.

[64] J. Wang, J. Wang, Forecasting stochastic neural network based on financial empirical mode decomposition, Neural Networks 90 (2017) 8–20.

[65] J. S. Armstrong, Combining forecasts: The end of the beginning or the beginning of the end?, International Journal of Forecasting 5 (4) (1989) 585–588.

[66] S. Gupta, P. C. Wilton, Combination of forecasts: An extension, Management Science 33 (3) (1987) 356–372.

[67] C. Christodoulos, C. Michalakelis, D. Varoutas, Forecasting with limited data: Combining arima and diffusion models, Technological forecasting and social change 77 (4) (2010) 558–565.

[68] J. H. Holland, Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence, MIT press, 1992.

[69] D. E. Golberg, Genetic algorithms in search, optimization, and machine learning, Addion wesley 1989.

[70] J. R. Koza, Genetic programming: on the programming of computers by means of natural selection, Vol. 1, MIT press, 1992.

[71] B. Wu, C.-L. Chang, Using genetic algorithms to parameters (d, r) estimation for threshold autoregressive models, Computational Statistics & Data Analysis 38 (3) (2002) 315–330.