+A Dynamic Neural Network Architecture with immunology Inspired Optimization for Weather Data Forcasting

Abir Jaafar Hussain[1,*], Panos Liatsis[2], Mohammed Khalaf[1], Hissam Tawfik[3], Haya Al-Asker[4],

[1]Applied Computing Research Group
Liverpool John Moores University
Liverpool L3 3AF, UK

[2]Khalifa University of Science and Technology
Department of Computer Science
Abu Dhabi, United Arab Emirates

[3]Leeds Beckett University
School of Computing, Creative Technologies & Engineering
United Kingdom LS1 3HE

[4]Salman Bin Abdulaziz University
Department of Computer Science,
Al-Kharj - PO Box 151 - ZIP code 11942,
Saudi Arabia

[*]Corresponding Author email: a.hussain@ljmu.ac.uk

Abstract—Recurrent neural networks are dynamical systems that provide for memory capabilities to recall past behaviour, which is necessary in the prediction of time series. In this paper, a novel neural network architecture inspired by the immune algorithm is presented and used in the forecasting of naturally occurring signals, including weather big data signals. Big Data Analysis is a major research frontier, which attracts extensive attention from academia, industry and government, particularly in the context of handling issues related to complex dynamics due to changing weather conditions. Recently, extensive deployment of IoT, sensors, and ambient intelligence systems led to an exponential growth of data in the climate domain. In this study, we concentrate on the analysis of big weather data by using the Dynamic Self Organized Neural Network Inspired by the Immune Algorithm. The learning strategy of the network focuses on the local properties of the signal using a self-organised hidden layer inspired by the immune algorithm, while the recurrent links of the network aim at recalling previously observed signal patterns. The proposed network exhibits improved performance when compared to the feedforward multilayer neural network and state-of-the-art recurrent networks, e.g., the Elman and the Jordan networks. Three non-linear and non-stationary weather signals are used in our experiments. Firstly, the signals are transformed into stationary, followed by 5-steps ahead prediction. Improvements in the prediction results are observed with respect to the mean value of the error (RMS) and the signal to noise ratio (SNR), however to the expense of additional computational complexity, due to presence of recurrent links.

Keywords— Recurrent Neural Networks, Immune Systems Optimisation, Time Series Data analytics, weather forecasting.

1. Introduction

In the past two decades, significant improvements and the evolution of big data in weather forecasting attracted researchers to the big data domain mainly due to the large amount and variety of data that need to be handled [1]. Scientists are working with Big weather Data, characterised by complexity at one or more of the main 5 Values (5Vs) [2]. Big Data techniques need to store, process, and mine weather applications information in an effective and efficient manner to generate information that can improve the accuracy of weather prediction. The challenge in this field is to provide accurate predictions about weather status, in the context of challenges with regards to handling, processing and extracting valuable information from very large and complex weather data.

Time series analysis generally refers to a sequence of data points, measured typically in successive times, and spaced at regular time intervals. In practice, it is a collection of historical data of a system, such as the price of a stock, traffic data, and pollution rates [3-8]. A time series can be used in two ways with different objectives:

- Looking backward – the use of historical data to analyse the previous behaviour of a system. Applications include diagnosis or recognition of machine faults [9] or human disease [10, 11].
- Looking forward – the use of data to predict or forecast the future behaviour of a system. Applications include stock or price prediction [12], market demand forecasting [13] and weathering forecast [14].

Time series usually contain a component associated with random variations. Analysis of such data is a challenging task considering the variety of internal and external factors affecting a dataset. In their theoretical analysis, Herrera assumed that a time series is generated by a dynamical system [15]. Systems that generate time series possess complex properties, where the relationship between the elements of the time series is nonlinear and includes extensive dynamical behaviour. These properties make it difficult to accurately analyse the behaviour of such systems even when the underlying properties are completely known. Time series analysis has essentially helped in the development of both traditional and intelligent methods. Traditional methods require assumptions about the characteristics of the data. Intelligent techniques are based on training paradigms, which learn the behaviour of the time series. Analysis of time series behaviour of complex signals such as the ones related to the human body, stock markets, weather signals or even country economies is a major challenge. The main advantage of using intelligent methods based on machine learning techniques is the ability to perform with little or no prior information about the time series.

Machine learning is considered a field of science, aiming specifically at learning and extracting knowledge from data sets, in order to develop real world simulations, apply prediction,classification, and pattern recognition methodologies on the input data [16]. Among the several machine learning models in the sub-field of data prediction, neural networks techniques are effective and useful alternatives to statistical methods. The prediction process is used to detect values or events that have high probability to occur in the future. For many decades, artificial neural networks (ANNs) have been successfully used in prediction applications with remarkable levels of performance. The main objective of this empirical study is to build a dynamic neural network architecture, optimized using the immune system algorithm for forecasting of big weather data.

ANNs have been prevalent in most machine learning applications. The ‘popular’ multilayer perceptron (MLP) suffers from difficulties such as the determination of the optimal architecture and the values of the optimal weights. These parameters are important in the performance of the neural networks. Furthermore, the MLP is affected by some well-known learning problems, such as over-fitting [17-19]. This means that the neural network can perfectly perform the mapping between the inputs and outputs in the training data, however it will not be able to sufficiently generalize this performance to an unseen data set. There is a number of studies, which investigated possible methods to improve the generalization ability of feed-forward neural networks and automatically select the best number of hidden units and their weights. Widyanto et al. [19] proposed a new technique using a self-organized hidden layer inspired by the immune algorithm (SONIA). SONIA was used to predict temperature-based food quality, demonstrating an improvement of 18% when compared to MLPs [19].

However, SONIA is applicable to feed-forward neural networks, which means that it can solve static mapping problems, however it is not able to recall past behaviours and as a result it cannot produce a high performance in dynamical temporal data [20]. Subsequently, SONIA was extended to handling recurrent links in the output layer thus enabling its application in regression problems through the efficient processing of the temporal patterns present in the time series signals. The main advantage of recurrent connections in a neural network is their ability to deal with both static and dynamical situations [21, 22]. These links may enable aspects from cognitive functions, such as memory association, in the classification and prediction of dynamical systems. The work of Makarov et al.[23] showed that recurrent networks could be used to support the learning process in both dynamic and static problems. Furthermore, it has been proved that using recurrent feedback links can improve the network’s ability to analyse time series that are generated by complex systems.

A new dynamic self-organized neural network inspired by the Immune Algorithm is proposed in this work. The network consists of three layers. The first layer accommodates for the input data and previous output values. The hidden layer is created using the self-organized learning rules based on the Immune Algorithm, while the output layer holds the output values of the forecasted signals. The rationale behind the use of recurrent links from the output to the input layers is to improve the prediction and generalization ability of the network by providing a memory feature of past behaviours. This is achieved in the expense of computational complexity and having to resolve the stability issues associated with the use of recurrent links. The main contribution in this paper is the design and application of the dynamic self-organized multilayer neural network inspired by the immune algorithm (DSMIA) for the prediction of weather signals. This is employed to address the complexity, high-volume, and non-linear nature of big weather data signals, using neural computing techniques with a view to gaining optimal weather forecasting outcomes.

The reminder of this paper is organized as follows. Section 2 discusses Big Weather Data and associated challenges, while section 3 provides an overview of the self-organized neural network inspired by the immune algorithm (SONIA). Section 4 introduces the proposed dynamic self-organized multilayer neural network inspired by the immune algorithm (DSMIA). The methodology for the experiments in this contribution is presented in section 5, while section 6 presents the simulation results. Finally, section 7 is dedicated to the conclusions and future directions of this research.

2. Big 'Weather' Data and Challenges

Over the last two decades, Big Data has become an important and primary knowledge discovery approach for large-scale datasets in many domains [24-26]. Big data typically comprises datasets with sizes beyond the capability of frequently utilised software platforms to process, capture, manage and curate within tolerable scales [27]. This field can be divided into three application areas, namely, structured, semi-structured and unstructured data; therefore, one of the main concerns is how to understand and process unstructured data. This requires a set of technologies and methods that can reveal insight about datasets that are complex, diverse, and of massive scale [27]. With the large amount of data available today, Big data analytics techniques promise to offer transformative potential and opportunities for advances in several areas, including weather forecasting. As the size of data keeps on getting bigger, machine learning techniques including ANNscan play a crucial role in provide optimal solutions and suggestions in big data predictive analytics [28, 29].

Zikopoulos et al. (2012), describe 'Big Data' as consisting of a set of three main 'V-words', i.e., volume, velocity and variety, and two additional Vs, i.e., veracity and value [30]. In what follows, we will briefly discuss the 5 Vs of big data [27, 31]:

- **Volume**: This category refers to the total size of data, which used to be measured in Gigabytes and currently measured in Yottabytes (YB). The data size determines the potential and value insight to be considered big data or otherwise. The size of data that comes from various sources is presenting a huge challenge, which renders old-style database technology inappropriate in storing, collecting and analysing of big data. What is instead required is advanced distributed management systems, where parts of the data are stored in several warehouses and can be accessed through special software such as Hadoop. Ismail et al [32] proposed a new prediction framework for Big Data analytics based on the Hadoop MapReduce algorithm. Hadoop is considered an effective platform that offers efficient functionalities for processing and storing of large amounts of data. In relation to volume, every day, the meteorological weather department receives a hug amount of weather data sets from various sources. The analysis of such big weather datasets is considered a major research challenge.

- **Velocity**: The second characteristic of big data refers to the speed in which data is produced, collected, processed and analysed. Big data technology currently permits the analysis of large volume of data, as it is being produced in real-time, without the need to transfer it into database warehouses. In the example of weather forecasting, Radar observations play an increasingly significant role in weather prediction, where real-time forecasts of actual storms, initialized by current data, are within reach [33].
- **Variety**: This aspect refers to the various types of data that are being produced. Typically, the vast majority of weather data is unstructured and difficult to be tabulated or categorised. In the weather domain under consideration, there are various sources that produce big weather data, such as sensors, satellite images, and information about solar lightintensity.
- **Veracity**: This kind of Big Data characteristic is considered one of the biggest challenges. It refers to the trustworthiness or messiness of the data which can be related to various forms of data abnormalities and imperfections. Organizing weather data in a meaningful manner is not an easy task, particularly when the data itself changes quickly.
- **Value**: It is paramount that significant value or payoff can be discovered in big data. Accurate weather predictions have been established as offering high value and having various applications, e.g., in agriculture, energy efficiency, natural disaster management, etc.

In summary, Big Data deals with the analysis and management of huge amounts of data with highly varying dynamics, characterized by complex structure [2]. Moreover, this field has the potential for major advances so as to reduce data redundancy, and speed up access, distribution of stored data and improve their availability [34]. The field of Big Data is one of the most discussed topics in the state of the art, and this trend is predicted to continue in the future [35, 36]. In [37], it is stated that Big Data is “a collection of data with complexity, diversity, heterogeneity, and high potential value that are difficult to process and analyse in reasonable time”.

The main challenges facing Big Data in the weather forecasting domain are due to the nature of its very large volume, speed and inherently complex underlying behaviour and characteristics [38]. Weather data changes and develops in real time, and it is essential that, the methods utilised for weather forecasting can accurately produce dynamic predictions.

2.1 Related works

The development of technology for weather forecasting has played an important role in the environmental domain [39]. The aim of these developments is to improve the utilisation of technology in the environmental community, particularly in the area of weather forecasting. Expert systems and various Artificial Intelligence (AI) techniques have been used and developed to improve decision support tools, e.g., in flood management [40]. Machine Learning models (ML) are considered to be powerful techniques in the field of scientific research that enable computers to learn from data [16, 41]. ANNs have been particularly popular in this application area.

ANNs consists of elementary processing units, known as neurons, which are grouped in layers and are interconnected, via weights, so as to form network structures [42, 43]. The weights of a neural network are trained using a training algorithm, which could be based on supervised or unsupervised learning. In supervised learning, a target output is used to modify the weights of the network, through an error minimization process, for a specific set of input patterns. Unsupervised learning on the other hand does not require a target output, but rather exploits correlations in the input data.

Extensive researches have indicated that dynamic neural network architectures generate significant improvements when used in the pre-processing of weather forecasting time-series data signals and have assisted in obtaining a high degree of accuracy in the prediction of weather data sets [44-47]. Grimes et al [48] proposed a model, where Cold Cloud Duration (CCD) imagery derived from Meteosat thermal infrared imagery is used in integration with numerical weather analysis data as input to an artificial neural network. Principal component analysis (PCA) is used to reduce the data dimensionality in weather analysis in addition to a pruning method which recognises redundant input data. The original dataset contains rain gauge data from central Africa collected over a period of 4 years. Calibration and validation were conducted by employing rainfall estimation data from the daily rain gauge data. The neural network approach demonstrated higher prediction accuracy, when compared to the traditional CCD model.

Mandale and Jadhawar developed an efficient technique based on Data Mining for weather forecasting [49]. The study was conducted using Decision Tree Algorithms and ANN. In order to classify the data sets, rainfall, maximum temperature, minimum temperature, wind speed, and evaporation parameters were used as the main weather inpout data in this study to predict future weather conditions. The performance of this approach was evaluated using metrics such as the correlation coefficient, the mean squared error, and the normalised mean squared error.

The massive accessibility of weather forecasting data in the last decades, such as radar, satellite maps, and observational records requires increased attention in order to find an effective platform to analyse and extract hidden knowledge embedded in big data. Dutta et al [50] applied data mining method for forecasting rainfall in the region of Assam on a monthly basis. The original data sets were collected from the Regional Meteorological Centre in a six-year period between 2007 to 2012. The MLP network and multi-linear regression were applied to this forecasting problem. In order to verify the performance and accuracy of the proposed approach, cross-validation was used. The performance was measured using the adjusted R square, and the mean squared error metrics.

in summary, the main purpose of collecting, processing, and storing weather data is to provide accurate prediction for weather trends. Various types of sensors are used by meteorological departments for data collection, such as humidity, temperature, and water level sensors. The overview of the state of the art demonstrates that current contributions are still limited and further investigation is still required [48, 49]. The aim of the current study is to demonstrate the superior performance of the proposed dynamic neural networks architecture in weather prediction, using a variery of performance metrics, such as the Normalised Mean Square Error (NMSE), Mean Square Error (MSE), Mean Absolute Error (MAE), and Signal Noise Ratio (SNR). The case study addressed in the current contribution involves the prediction of three important features of weather forecasting. These are valley sunshine, valley max temperature, and valley rainfall. These features can provide significant support to meteorological departments in the context of weather status prediction. For comparison purposes, we use various neural network architectures to investigate their accuracy and performance on the problem at hand.

3. Self-organised Network Inspired by the Immune Algorithm (SONIA)

The immune system is a biologically inspired pattern recognition and classification system [51]. By observing the mechanisms of immune systems occurring in biological beings, researchers identified many interesting processes and functions, which can provide useful metaphors for computation. The Artificial Immune System (AIS) algorithm can learn to distinguish the ‘self’ from ‘non self’ and solve relevant classification problems [52]. Additionally, naturally occuring immune systems perform important maintenance and repair functions [53]. Artificial immune systems are one of the most rapidly emerging biologically motivated computing paradigms. There is significant growth in the applications of immune system models across many fields [54]. Examples include computer security, function optimization, control engineering, data mining, pattern recognition, image interpretation, anomaly detection, sensor fusion, and process monitoring [55].

The Artificial Immune Recognition System (AIRS) is an immune system inspired supervised learning algorithm [56]. It uses immune mechanisms of resource competition, clone selection, maturation, mutation and memory cells generation. The training and test data items are viewed as ‘antigens’ in the system. These antigens induce the B-cells in the system to produce artificial recognition balls (ARBs), which compete for the given resource number. ARBs with higher resources get more chances of producing mutated offspring to improve system performance. Memory cells generated after all training antigens are introduced are subsequently used to classify test data.

The SONIA network [19] is a single hidden layer neural network, which consists of a self-organising hidden layer, optimized through the use of the immune algorithm and an output layer trained using the traditional back-propagation algorithm. The immune algorithm is simulated as the natural immune system, which is based on the relationship between its components, which involve antigens and cells; this is called a recognition ball (RB). The recognition ball in the immune system consists of a single epitope and many paratopes, where the epitope is attached to the *B* cell, and paratopes are attached to antigens [57]. The *B* cell here represents several antigens. In the context of biology, a *B* cell can be created and mutated to produce a diverse set of antibodies to remove and fight viruses attacking the body. Thus, the immune system can allow its components to change and learn patterns by changing the strength of connections between individual components. In the case of the SONIA network, the input units are called antigens and the hidden units are considered as the recognition ball (RB) of the immune system. The input vector represents an antigen, while the hidden layer of the network is considered as a recognition ball as shown in Fig. 1. The recognition ball is used to create hidden units. The relation between the antigens and the RB is based on the definition of local pattern relationships between input vectors and hidden nodes. These relationships help SONIA to easily recognise and define the input data’s local characteristics, which increases the network’s ability to recognise patterns. In SONIA, the mutated hidden nodes are designed to deal with unknown data, i.e., test data, so as to enhance the generalisation ability of the network.

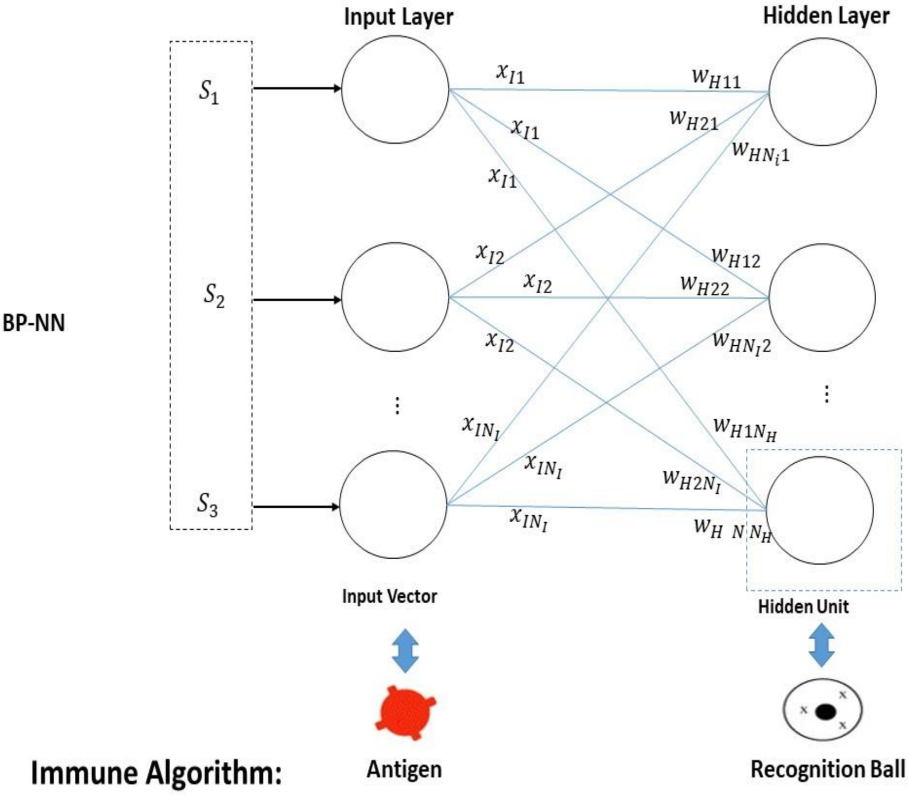


Fig. 1. Input vector and hidden units of the Backpropagation-NN are considered as antigen and the recognition ball of the immune algorithm, respectively [18].

4. Dynamic Self-Organised Multilayer network Inspired by the Immune Algorithm (DSMIA)

In this research, we build on past work and propose a new dynamic neural network architecture that incorporates recurrent links within its structure to create a self-organising layer, inspired by Artificial Immune System theory [53]. In short, recurrent links are introduced in the SONIA network architecture, thus allowingthe capture of complex patterns found in the natural time series.

The proposed network has three layers, the input layer, the hidden layer and the output layer, as illustrated in Fig. 2. It includes the dynamic self-organisation of hidden-layer units, and feedback links to the input layer. As such, the previous behaviour of the network is used as an input affecting the current behaviour. Similar to the Jordan recurrent network [9], the output of the network is fed back to the input through the context units. This represents a major improvement compared to feed-forward networks, which can only implement a static mapping of the input vectors. In order to model dynamic functions, it is essential to exploit the structure of a system capable of storing internal states and implementing complex dynamics. Neural networks with recurrent connections are dynamic systems with temporal/state representations, which, because of their dynamic structure, have been successfully used in solving a variety of problems.

This section provides an overview of the Dynamic Self-Organising Multilayer network, inspired by the Immune Algorithm (DSIA), as shown in Fig. 2.

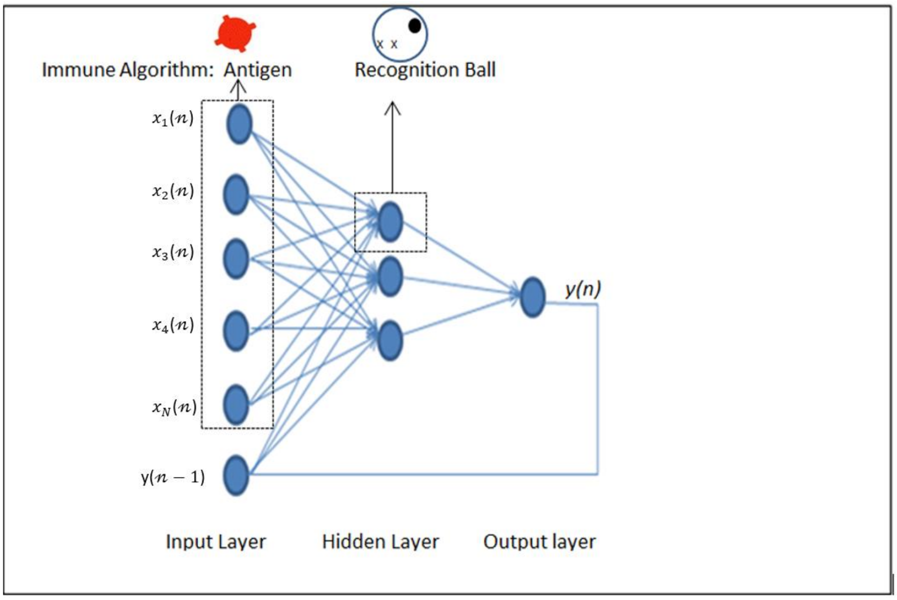


Fig. 2. The structure of DSMIA network.

In the self-organising Kohonen networks (SOM), each unit *j* of a map (1<=*j*<=*nh*), where *nh* is the number of hidden units, is compared with the weight vector w_j and an input *x(t)* and *t=1,…,ni*, and *ni* is the number of input units and the output. The Euclidean distance function is used for the comparison between w_j of the hidden map and input $x(t)$:

$$E_j = \sqrt{\sum_{i}^{ni} (x(t)_i - w_{ji})} \tag{1}$$

For an input vector, the best matching unit is the unit that minimizes the error function:

$$E = \left\| x(t) - w_j \right\| \tag{2}$$

The learning rule is based on updating the weights of neurons that are related to a neighbourhood of the best matching unit:

$$\Delta w_j = \gamma h_k (x(t) - w_j) \tag{3}$$

where γ is the learning rate, k is the index of the best matching unit and h is the neighbourhood function, which decreases the distance between units j and k on the map.

Suppose that N is the number of external inputs $x(n)$ to the network, and $y_k(n-1)$ is the output of the network from the previous time step $(n-1)$ and let O represents the number of outputs. In the proposed *DSMIA*, the overall input to the network will be the union of the components of $x(n)$ and $y_k(n-1)$ and thus, the number of inputs to the network is $N+O$ defined as U where

$$U(n) = \begin{cases} x_i(n) & i = 1, \dots, N \\ y_i(n-1) & i = 1, \dots, O \end{cases} \tag{4}$$

The output of the hidden layer is computed as

$$v_{hj}(n) = \alpha \sqrt{\sum_{i=1}^{N} (w_{hji} - x_i(n))^2} \tag{5}$$

$$z_{hj}(n) = \beta \sqrt{\sum_{k=1}^{O} (wz_{hjk} - y_k(n-1))^2} \tag{6}$$

$$D_{hj}(n) = v_{hj}(n) + z_{hj}(n) \tag{7}$$

$$x_{hj}(n) = f_{ht}\,(D_{hj}(n)) \tag{8}$$

$$\hat{y}_k = f_{ot}\left[\sum_{j=1}^{N_H} w_{ojk}\, x_{Hj} + b_{ok}\right] \tag{9}$$

where f_{ht}, f_{ot} are nonlinear activation functions for the hidden and output layers, respectively, N is the number of external inputs, b_{ok} is the bias term, O is the number of output units. w_{hj} are the hidden layer weights corresponding to the external inputs, while wz_{hjk} are the hidden layer weights associated with the previous outputs, w_{ojk} are the output later weights, b_{ok} is the bias of output unit k, n is the current time step, while α, β are user-selected parameters with $\alpha>0$ and $\beta>0$.

The first layer of the DSMIA is a self-organised hidden layer trained similarly to the recursive self-organized map (RecSOM) [58]. In this case, the training rule for updating the weights is inspired by the use of the immune algorithm in the SONIA network [17]. However, in DSMIA, the weights of the context nodes wz_{hjk} are updated in the same way as the weights of the external inputs w_{hj}. This is done by first finding D_{hj}, which is the distance between the input units and the centroid of the j^{th} hidden unit:

$$D_{hj}(n) = \alpha \sqrt{\sum_{i=1}^{N_i} (X_{i(n)-} W_{hji}(n))^2} + \beta \sqrt{\sum_{k=1}^{N_o} (y_{k(n-1)-} W_{zjk}(n))^2} \tag{10}$$

From $D_{hj}(n)$, the position of the closest match is determined as:

$$D_c(n) = argmin\{D_{hj}(n)\} \tag{11}$$

If the shortest distance D_c is less than the stimulation level value, $s_1 \in (0, 1)$, then the weights of the external input vectors and the context vectors are updated as follows:

$$w_{hji}(n+1) = w_{hji}(n) + \gamma D_c(n) \tag{12}$$

$$wz_{hji}(n+1) = wz_{hjk}(n) + \gamma D_c(n) \tag{13}$$

where wz_{hji} are the weights of the previous outputs and w_{hji} are the weights of the external inputs, and γ is the learning rate, which is updated during the epochs.

The purpose of hidden unit creation is to form clusters from the input data and to determine the centroid of each cluster. The centroids are used to extract local characteristics of the training data and to enable the DSMIA network to memorise the characteristics of training data. The use of the Euclidean distance to measure the distance of the input data and these centroids enables the network to exploit local information in the input data, while the recurrent links enable recall ofpast behaviours.

4.1 Recurrent neural networks (RNN)

In the last couple of years, various weather applications based on RNN have been developed [59-61]. One of the most prominent applications of RNN is pattern recognition, such as in weather forecasting systems [62]. RNN form complex nonlinear decision boundaries and utilise memories of the internal state of the network, which is crucial in dynamic prediction and classification tasks [63-65]. A number of studies confirmed that RNN have the ability to discover both linear and nonlinear relations in weather data [66]. However, previous studies have undertaken the classification of sequence-oriented data, for which the dependencies between elements of data are exploited in learning. In this work, we intend to explore the use of RNN for pattern recognition tasks, where data elements are assumed to be independently drawn.

In addition, it has been shown that RNN have the ability to provide an insight into the features used to represent biological signals [67]. Therefore, the employment of a dynamic tool to deal with time series data predictions is highly recommended [68]. This type of neural network has a memory that is capable of storing information from past behaviours [69, 70]. One of the most important applications of RNN is in modelling and identifying temporal patterns. Chung et al. [71] provide a relevant commentary of this aspect, highlighting that "recurrent (artificial) neural network models are able to exhibit rich temporal dynamics, thus time becomes an essential factor in their operation". Different studies indicated that RNN can be applied to non-linear decision boundaries [63]. In addition, the main advantages of recurrent neural networks is their ability to deal with static and dynamical behaviours [23, 72]. One of their powerful capabilities is finite state machine approximation, which makes recurrent neural networks suitable for learning both temporal and spatial patterns [73]. This kind of network is very useful in real-time applications, such as weather signal processing and analysis.

In principle, RNN can utilise the feedback network connections to store representations of new input events in the form of activations as in the Long Short-Term Memory neural network (LSTM) [74]. LSTM are a special kind of RNN introduced by Hochreiter and Schmidhuber in 1997, which can be used for both classification and regression, with any compatible objective (e.g., MSE) and capable of learning long-term dependencies [74]. LSTM can solve many tasks compared to previous RNN learning models [75]. It is a particularly promising type of recurrent neural networks, which is capable of forming a ‘bridge’ with long delays between inputs and outputs, and thus enabling access to long range temporal context [76]. A number of researchers applied LSTM in solving a wide variety of real-world problems, such as speech recognition [77, 78], protein secondary structure prediction [79], handwriting recognition [80], and reinforcement learning [81]. One of the key features of the LSTM is its ability to identify between recent and early examples through the use of dedicated weights, which allow for forgetting memories which are irrelevant in predict the output [82]. Thus, it is a good candidate approach in tackling long sequence inputs, compared to other RNN architectures, which able to memorise short sequences.

The LSTM network is an RNN with a specialised kind of hidden layer(s) that uses memory gates to overcome the vanishing/exploding gradient problem which renders backpropagation over deep networks ineffective. It can be considered a form of deep learning, since the network is equivalent to a feed-forward network with many layers that share weights, hence the need to overcome the vanishing/exploding gradient issue. This approach has the ability to remember information, which remains unchanged for long periods. Moreover, another benefit of LSTM is the ability to determine the optimal time lag in time series problems [83]. The model involves one input, one output, and one forgetting gates. In contrast to the traditional neural network, the basic unit of the hidden layer is the memory block [84].

Zhang et al., [85] presented a new approach based on the LSTM neural network for prediction of sea surface temperature (SST), which provides short-term prediction on a one day basis, three days prediction, and long-term prediction, as inweekly mean and monthly mean basis. In this research, two types of LSTM networks are used: a FC LSTM and a LSTM layer. The FC LSTM is used to map the output of the LSTM layer. While the LSTM layer is used to tackle the time series connection. The SST anomaly data were used in testing the network's prediction accuracy.

Zaytar and El Amrani [59] proposed a novel deep neural network architecture (DLNN) for weather prediction and applied it in time series data sets. The model concentrates on multi stacked LSTMs in order to map sequences of weather parameters of the same length. The aim is to generate two kinds of models for each city in Morocco in regards to forecasting three important values, i.e., wind speed, temperature, and humidity. The time series data was collected in a period of 15 years and was used to train the classifier. The outcomes illustrated that LSTM based DLNN are more effective, compared to traditional methods.

As previously indicated, there are promising outcomes to be achieved using RNN, including LSTM. This approach was proved to be powerful and particularly effective in sequence labelling, thus presenting a promising alternative for tackling the weather forecasting problem considered in this research. However, one of the main limitations of using LSTM is that it only deals with one input, which limits its potential to minimise the expected error. The activation function in LSTM is considered more complex than in other ANN architectures. Several studies seem to also suggest that LSTM higher complexity thanother models [74, 86, 87], and this in turn could hinder the use of LSTM in the case of real-time problems with instant feedback, as in the case of the current weather time series prediction scenario. In this research, we apply instead the Dynamic Self Organized Neural Network Inspired by the Immune Algorithm, and the Elman and Jordan neural networks which offer promising opportunities to address the weather prediction time series problem.

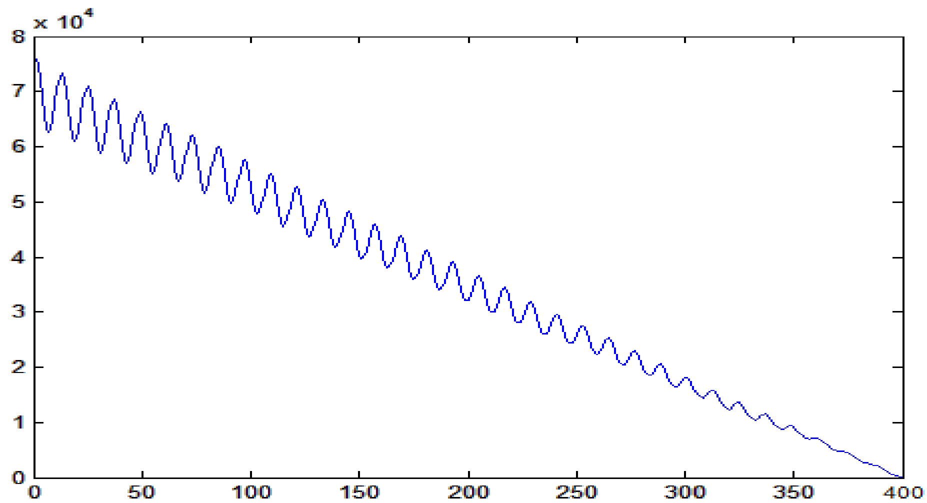
In particular, we propose a Dynamic Self Organized Neural Network Inspired by the Immune Algorithm, where learning focuses on the local properties of the signal and the aim of the network is to adapt to the local properties of the present data, while remembering past behaviours of the observed signal using the self-organised hidden layer inspired by the immune algorithm and recurrent links. Thus, the proposed network has the potential to offer a detailed mapping of the underlying structure within the data and is able to respond more readily to any significant changes, which common occur in non-stationary time series, such as the weather data.

5. Methodology

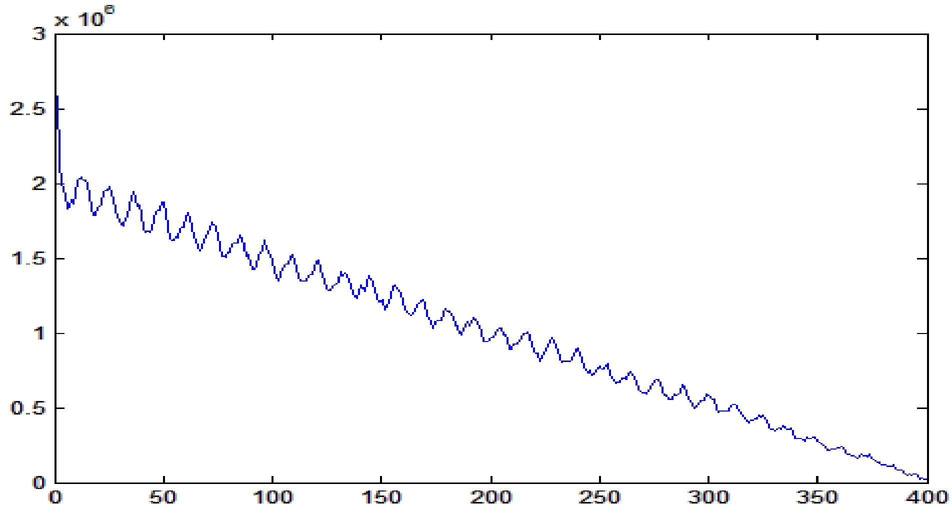
Three noisy weather time series are utilized in our experiments obtained from the Valley weather station in Anglesey (North Wales, UK). For experimental purposes, 400 points are selected for the prediction. They correspond to the period from November 1980 until February 2014 (per month) and represent the maximum temperature, rainfall in mm and sunshine in hours. As commonly encountered in practice, the three weather times series exhibit two distinct characteristics, i.e., nonlinearity and nonstationary. Fig. 3 shows that the correlograms of the three weather signals indicate that the signals are periodic, and the autocorrelation coefficient drops to zero for large values of the lag. Thus, we conclude that the time-series are nonstationary in nature.

The prediction performance of the neural networks is evaluated using four statistical metrics, which are used to provide accurate tracking of the signals as shown in Table 1. These include the Normalised Mean Square of the Error (NMSE), the Mean Square of the Error (MSE), the Mean Absolute value of the Error (MAE) and the Signal to Noise Ratio (SNR).

(a)



(b)



(c)

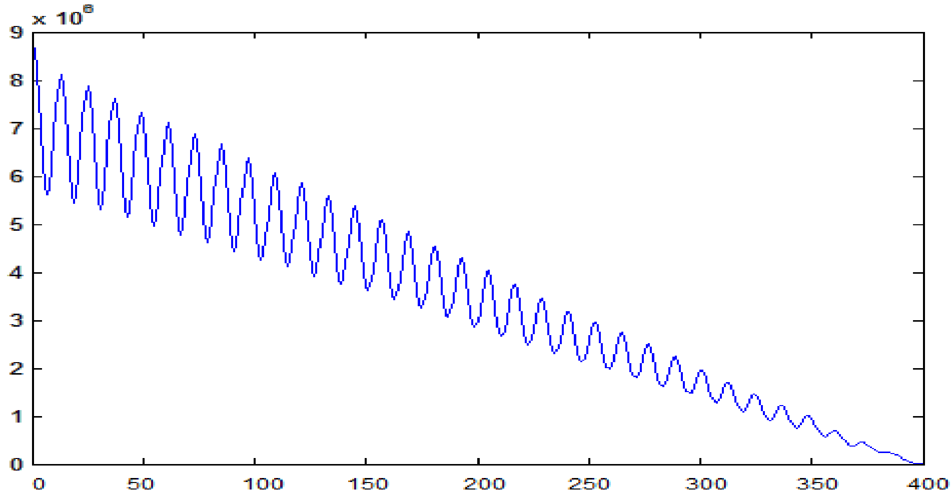
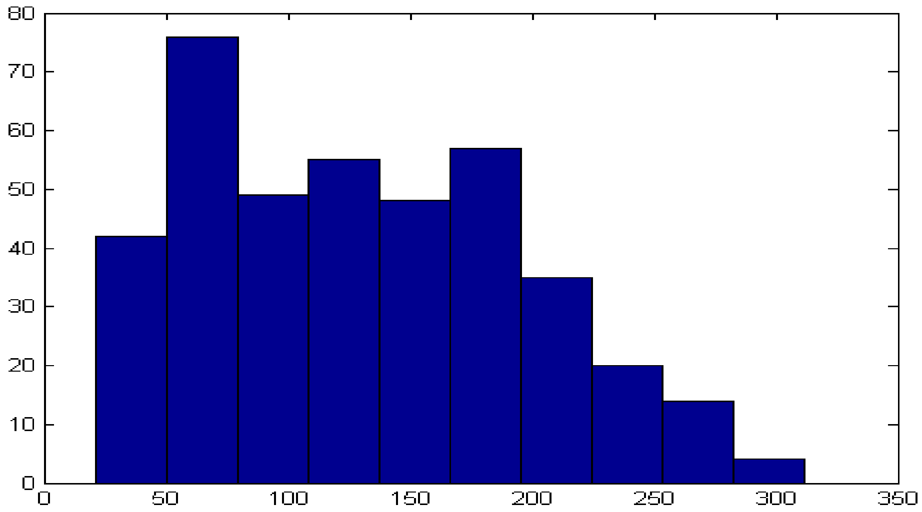


Fig. 3. The correlograms of the weather time series: (a) Valley temperature, (b) Valley rainfall and (c) Valley sunshine.

a)



b)

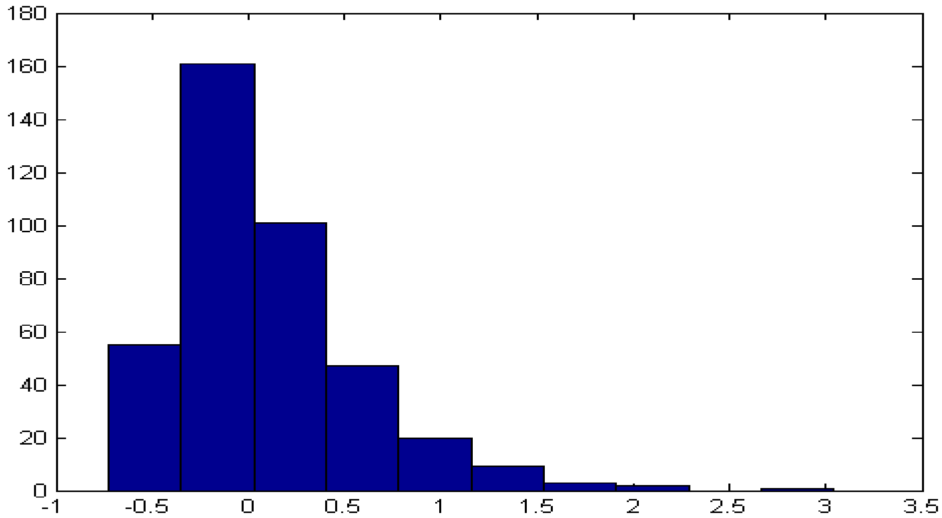


Fig. 4. The histogram of a) the original sunshine signal of the Anglesey valley, b) the transformed sunshine signal of the Anglesey valley.

Table 1
Performance metrics and formulae

Metrics	Calculations
NMSE	$NMSE = \frac{1}{\sigma^2 * N}\sum_{i=1}^{N}(y_i - \hat{y}\ )^2$ $\sigma^2 = \frac{1}{N-1}\sum_{i=1}^{N}(y_i - \bar{y}_i\ )$
MSE	$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}\ )^2$
MAE	$MAE = \frac{1}{N}\sum_{i=1}^{N}\lvert y_i - \hat{y}\rvert^{\,2}$
SNR	$SNR = 10 * \log_{10}(sigma)$ $sigma = \frac{m^2 * n}{SSE}$ $SSE = \sum_{i=1}^{n}(y_i - \hat{y}\ )^2$

	$m = \max(y)$

n is the total number of data patterns
y and $\hat{y}$ represent the actual and predicted output value

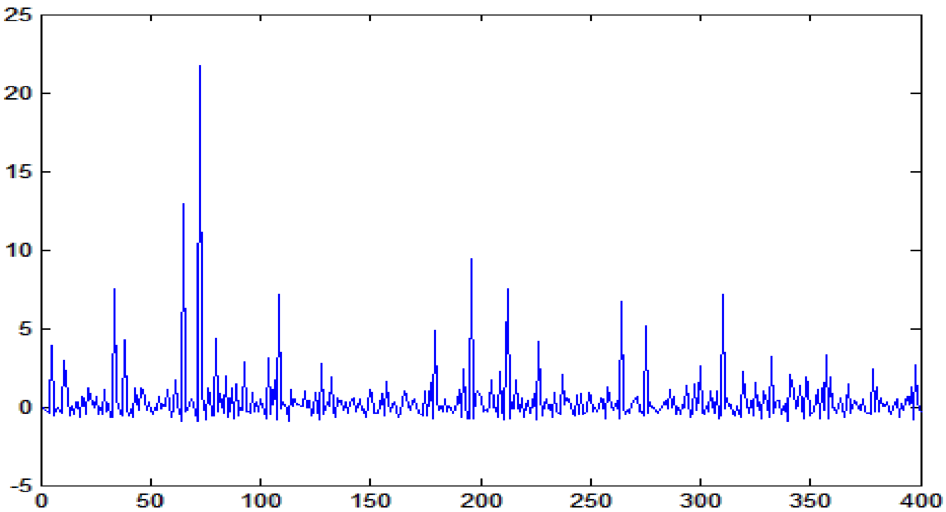
As previously discussed, the raw signals to be used in the experiments are non-stationary. Therefore, it is crucial to apply some pre-processing on the raw data before passing them to the neural network. The original non-stationary signals are transformed into stationary signals as follows:

$$R(n) = \frac{S(n)}{S(n-1)} - 1 \tag{14}$$

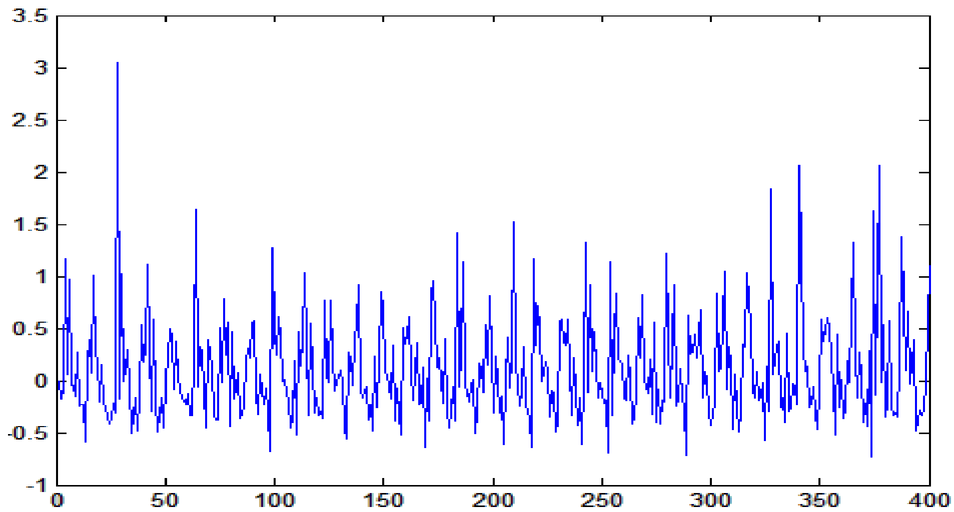
where *S(n)* is the input signal and *R(n)* is the one-step increased value at time *n*. This transformation has been shown to achieve better results [23]. *R(n)* has a relatively constant range of values even if the input data represent values over periods of many days, while the original data *S(n)* vary greatly, thus complicating the use of a valid model for long periods of time [24]. Another advantage of using this transformation is that the distribution of the transformed data becomes more symmetrical and resembles more closely the normal distribution.

Fig. 4 shows the histogram of the original sunshine signal of the Anglesey valley and its transformed signal. Although the transformed signal is not perfectly symmetric and does not accurately follow the normal distribution, the data is condensed more towards the zero bands. Fig. 5 shows that the original signals were transformed to stationary, indicated by less variation, hence improving the chances for better prediction.

(a)



(b)



(c)

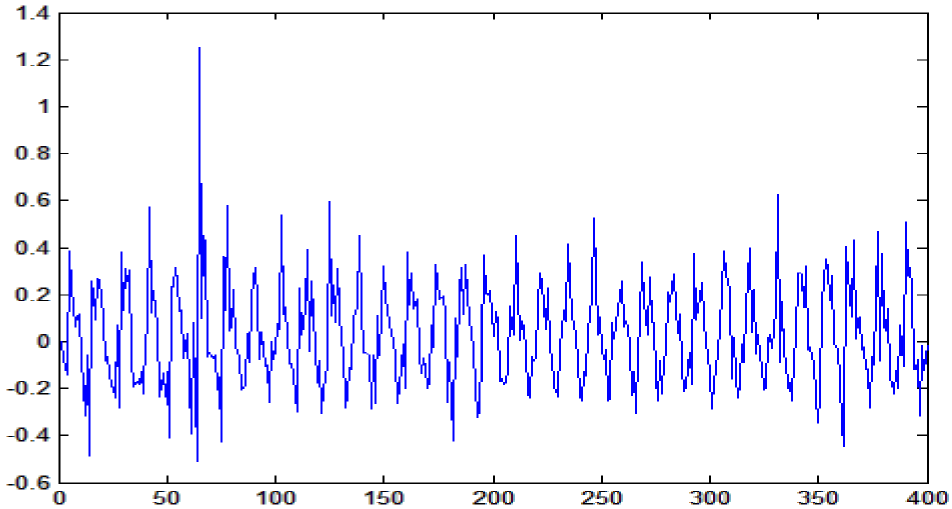


Fig. 5. The signals transformed to stationary: (a) Valley rain, (b) Valley sunshine, (c) Valley max temperature.

5.1 Experimental setup and environment

The experimental setup in this section covers the design of the test environment used in our experiments, the configuration of each model, and the models tested. Performance evaluation techniques were used to assess the results of the ANN in the weather datasets. The total proportion of the weather data set is divided into training and testing phases for evaluating the performance and generalisation ability. This method ensures that the generalisation error of the classifiers can be evaluated and also evaluates the capability of the neural networks to performance on unseen data.

In order to accommodate for dynamic links in the SONIA network, partially recurrent networks were used in this research. This type of recurrent neural network has feed-forward links as well as a selected set of feedback links. The feedback connections provides a memory to the network that will help the network to remember information from the past without excessively complicating network learning. Two different types of partially recurrent neural network topologies were utilised in order to develop new network architectures. The first type is the dynamic DSMIA, where the feedback links receive past data from the output layer. The second type is the dynamic DSIA, where the recurrent links receive past information from the hidden layer. In the next sections, the SONIA network and the two proposed networks are presented. The main motivation of these networks is to provide memory capabilities for the feed-forward self-organised network inspired by the immune algorithm.

In the case of DSIA, each unit j on the map has two weights, w_{hji} and W_{zj} , where w_{hji} are the weights linking the map with the input and W_{zj} are the weights linking the context unit, which is the output of the hidden layer at the previous time step, with the unit on the map:

$$D_{hj}\,(t) = \sigma(\alpha\,\|x\,(t) - w_{hji}\| + \beta\,\|X_{Hj}\,(t-1) - W_{zjj}\,\|\,) \tag{15}$$

$$X_{Hj}\,(t) = f_h\,(D_{hj}\,(t)) \tag{16}$$

where α >0 and β >0, || || denotes the Euclidean distance of vectors and f is a bipolar sigmoid function. The best matching unit is defined as the unit that minimises $D_{hj}\,(t)$):

$$c(t) = arrgmin\,\{D_{hj}\,(t)\} \tag{17}$$

$$c(t) = arrgmin\,\{D_{hj}\,(t)\} \tag{18}$$

Then the learning rule is applied to update the weights of input units and context units:

$$W_{hj}(t+1) = W_{hj}(t) + \gamma D_c\,(t) \tag{19}$$

$$W_{hj}(t+1) = W_{hj}(t) + \gamma D_c\,(t) \tag{20}$$

Where W_{zj} is the weight of the previous hidden unit and W_{hji} is the weight for the external inputs, and γ is the learning rate, which is updated during the epochs.

The results of the proposed DSMIA were benchmarked against state of the art neural network architectures. Figure 6 shows the proposed schematic for forecasting weather time series.

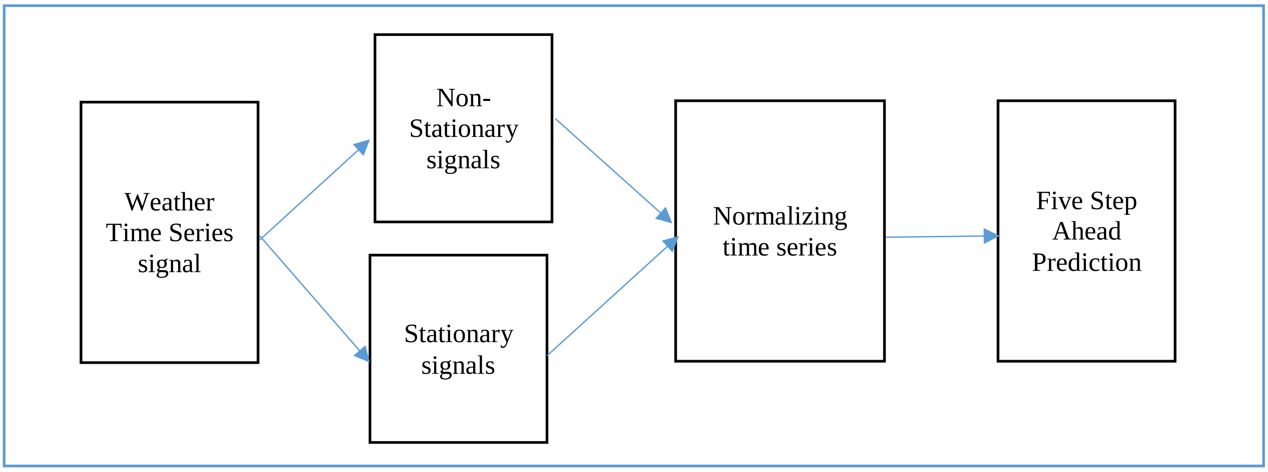


Fig. 6. Proposed framework for the prediction of weather time series

The main aspect of time series is that observation values are not created independently or ordered randomly; data in time series represents sequences of measurements arranged according to time intervals. Therefore, time variables are very important in time series analysis because they show when the measurements were recorded. Hence, [15] asserted that the time values must be stored along with observations that were recorded, and they should be used with the time series as a second piece of information. Therefore, the model that will be used to fit and analyse the time series data must have the ability to process the temporal pattern of the time series. Two main features characterise the time series data concepts. It is important to identify these two concepts before time series analysis, as this will assist in finding the best mathematical model to deal with this type of data. The simplest way to observe stationary and nonstationary data is the plotting of the observations. The concept of stationarity in time series means that the probability distribution between data does not change when shifted in time. Hence, the statistical properties (e.g., mean, variance and autocorrelation) of the data are stable with respect to time [25], such as climate oscillations [26]. In mathematics, stationarity can be defined as follows, when the distribution of $(x(t_1)$, . . ., $x(t_n))$ is the same as the distribution of $(x(t_{1+k})$, . . ., $x(t_{n+k})$ $x(t_{k+1}))$, where t_1, . . ., t_n refers to time step, and k is an integer. The behaviour of any intervals in this series is like one another, even if the segments have been taken from the beginning of the time series or the end. In order to apply multi-steps ahead prediction, two approaches could be pursued, namely direct and recursive. One-step prediction is carried out by utilising one or a number of measured past values. Direct multi-steps ahead prediction may utilise past values that were measured, as shown in Fig. 7. Recursive multi-step prediction can be used, y when a number of values are needed to be calculated. In other words, recursive prediction uses predicted values, rather than measured past values.

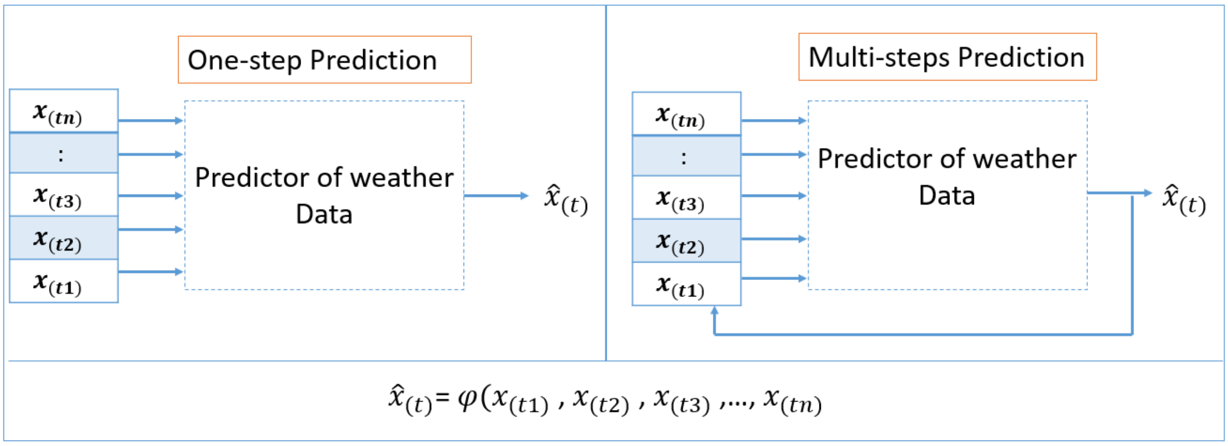


Fig. 7. One-step and multi-steps prediction of weather data

To evaluate the performance of the proposed models, we conducted a series of empirical simulations, using the previously described weather time series data setsWe provide full details of our analytical parameters in the methodology section, where aspects of the procedural setup are presented.

6. Simulation results

The simulation results for the prediction of the weather time series using the proposed DSIA network are presented in this section. The performance of the DSIA neural network is compared to the following non-hybrid Neural computing algorithms for predictors:

- Traditional MLP network
- Recurrent Elman neural network [51]
- Recurrent Jordan neural network [52]
- Feedforward SONIA neural network

Figs. 8, 9 and 10 show the original and predicted signals for the maximum valley temperature, valley sunshine, and valley rainfall using the DSMIA network for stationary data using 5 steps ahead prediction, respectively. The main parameters that been used in our experiment to estimate the predictions are NMSE, MSE, MAE, and SNR.

Tables 2 to 6 illustrate the average results for 30 simulations for the stationary prediction of the MLP, Elman, Jordan, SONIA, and the proposed DSMIA networks, respectively.

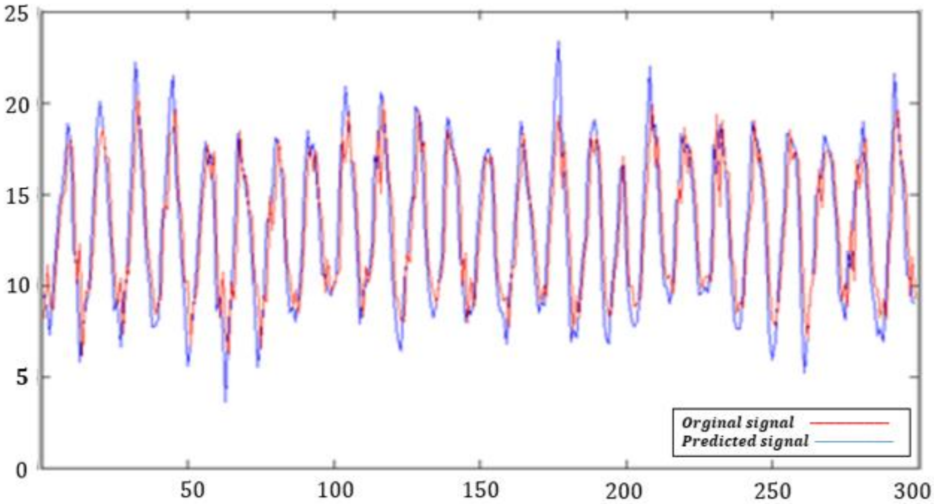


Fig. 8. Five steps ahead prediction for stationary maximum valley temperature signal using the DSMIA network.

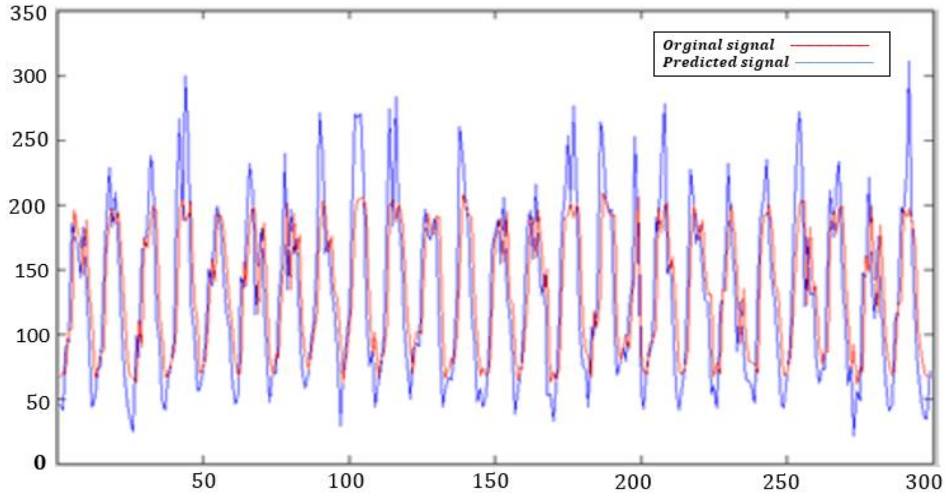


Fig. 9. Five steps ahead prediction for stationary valley sunshine signal using the DSMIA network.

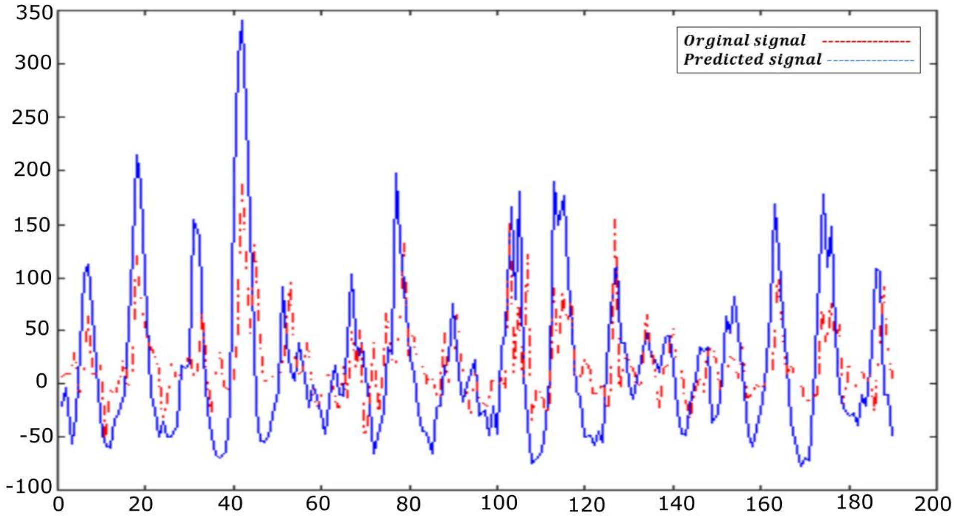


Fig. 10. Five steps ahead prediction for stationary valley rainfall signal using the DSMIA network.

Table 2

MLP networks simulation results for five steps ahead prediction

Signal	NMSE	MSE	MAE	SNR
Valley sunshine	0.91987	0.007105	0.06127	17.65
Valley max temp	0.70530	0.003736	0.05018	19.63
Valley rainfall	1.027654	0.0009102	0.02052	22.75
Average	**0.8843**	**0.0039**	**0.0440**	**20.0100**

Table 3
Elman networks simulation results for five steps ahead prediction

Signal	NMSE	MSE	MAE	SNR
Valley sunshine	1.1306	0.0087	0.0684	16.7721
Valley max temp	1.3881	0.0073	0.0646	17.2620
Valley rainfall	1.9227	0.0017	0.0284	20.5172
Average	**1.4805**	**0.0059**	**0.0538**	**18.1838**

Table 4
Jordan networks simulation results for five steps ahead prediction

Signal	NMSE	MSE	MAE	SNR
Valley sunshine	0.8359	0.0064	0.0588	18.06
Valley max temp	0.5707	0.0030	0.0434	20.55
Valley rainfall	1.3011	0.0011	0.0244	21.97
Average	**0.9026**	**0.0035**	**0.0422**	**20.1933**

Table 5
SONIA networks simulation results for five steps ahead prediction

Signal	NMSE	MSE	MAE	SNR
Valley sunshine	0.454872	0.011113	0.085950	13.16
Valley max temp	0.3512	0.005509	0.060488	20.18
Valley rainfall	1.0014	0.000887	0.020383	22.87
Average	**0.6469**	**0.0058**	**0.0556**	**20.2167**

Table 6
DSMIA network simulation results for five steps ahead prediction

Signal	NMSE	MSE	MAE	SNR
valley sunshine	0.200705	0.00415	0.048618	21.28
Valley max temp	0.062739	0.001527	0.029230	25.89
Valley rainfall	0.736493	0.007573	0.065009	17.17
Average	**0.3333**	**0.0044**	**0.0476**	**21.4467**

As it can be seen from Tables 2 to 6, the proposed DSMIA offered better results than the SONIA network for most of the weather signals using the NMSE and the SNR measures. This clearly indicates that the recurrent links provided the network with memory and hence better prediction with an average improvement of 1.23 dB in terms of the SNR. Furthermore, the proposed network shows slightly improved results than all the benchmarked networks.

Further experiments were conducted using the nonstationary weather data. Tables 7 to 11 show the average results for 30 simulations for the nonstationary prediction using the MLP, Elman, Jordan, SONIA, and the proposed DSMIA networks, respectively. While Figs. 11, 12 and 13 show the original and predicted signals for the maximum valley temperature, valley sunshine, and valley rainfall using the DSMIA network for nonstationary data prediction.

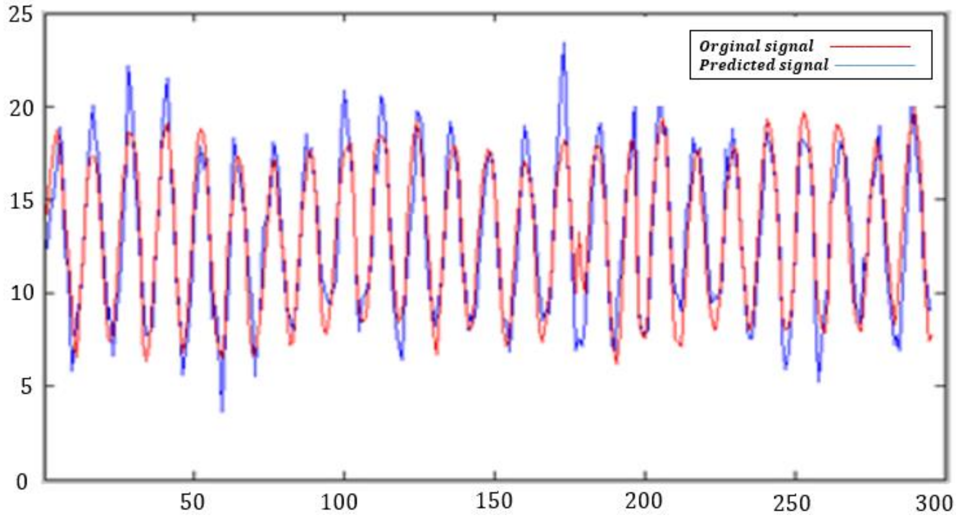


Fig. 11. Five steps ahead prediction for the non-stationary maximum valley temperature signal using the DSMIA network.

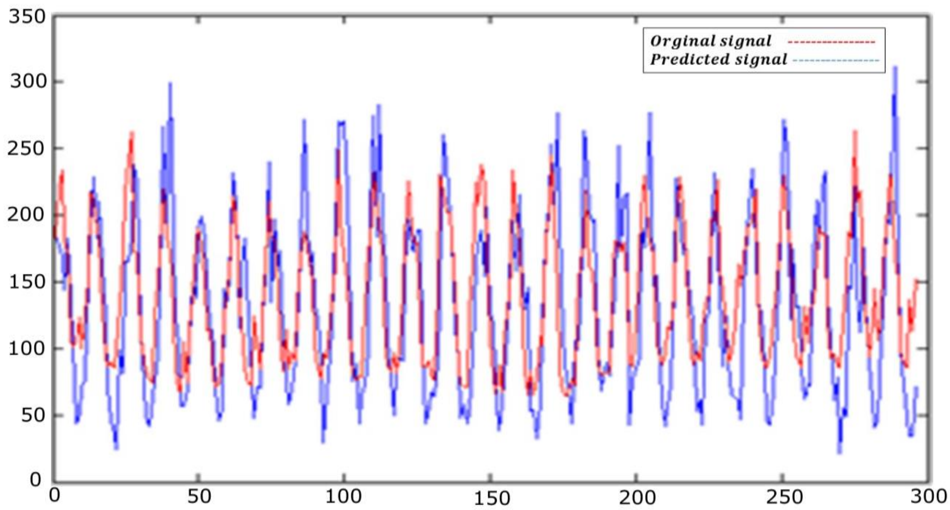


Fig. 12. Five steps ahead prediction for the non-stationary valley sunshine signal using the DSMIA network.

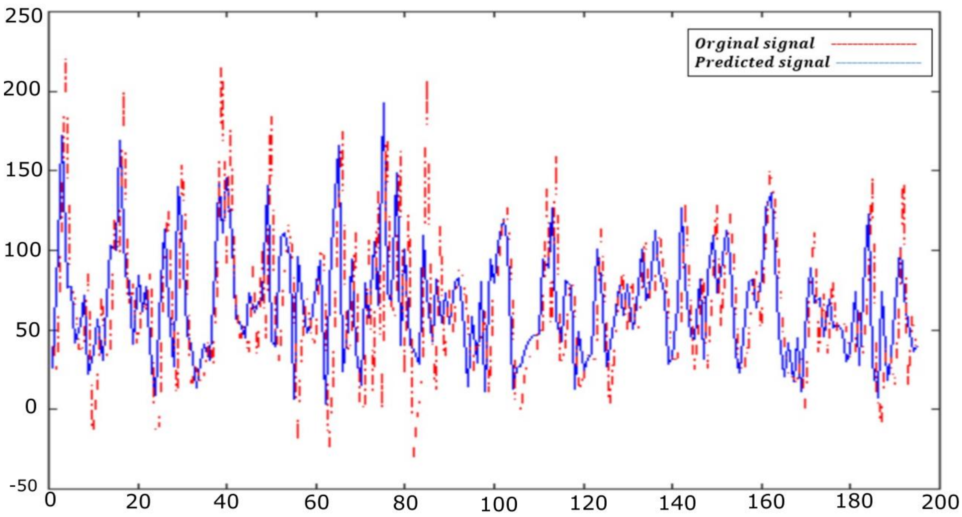


Fig. 13. Five steps ahead prediction for the non-stationary valley rainfall signal using the DSMIA network

Table 7
Simulation results for the MLP network in five steps ahead prediction for the non-stationary signals

Signal	NMSE	MSE	MAE	SNR
valley sunshine	0.8085	0.0153	0.0889	16.2208
Valley max temp	0.3877	0.0061	0.0626	19.7504
Valley rainfall	0.9289	0.0107	0.0808	17.3117

Table 8
Simulation results for the Elman network in five steps ahead prediction for non-stationary signals

Signal	NMSE	MSE	MAE	SNR
valley sunshine	0.6770	0.0128	0.0814	18.2988
Valley max temp	1.0950	0.0172	0.0945	17.5980
Valley rainfall	1.1137	0.0129	0.0892	16.5567

Table 9
Simulation results for the Jordan network in five steps ahead prediction for non-stationary signals

Signal	NMSE	MSE	MAE	SNR
valley sunshine	0.3410	0.0620	0.0064	19.9732
Valley max temp	0.1663	0.0026	0.0396	23.4705
Valley rainfall	1.8175	0.0210	0.1086	15.8251

Table 10
Simulation results for the SONIA network in five steps ahead prediction for non-stationary signals

Signal	NMSE	MSE	MAE	SNR
valley sunshine	0.578262	0.010928	0.085222	17.68
Valley max temp	0.350782	0.005502	0.060503	20.18
Valley rainfall	0.985458	0.011383	0.083236	17.06

Table 11
The simulation result for DSMIA network for five steps ahead predication for non-stationary signal

Signal	NMSE	MSE	MAE	SNR
valley sunshine	0.5478	0.0824	0.0104	17.9119
Valley max temp	0.1481	0.0392	0.0023	23.9300
Valley rainfall	0.9690	0.0112	0.0838	17.1298

The Normalised Mean Squared Error (NMSE) shows the overall deviations between the predicted and measured values. NMSE is a useful measure because if a system has a very low NMSE, then it indicates that it is correctly identifying patterns. As it can be seen in Tables 7 to 11, the proposed DSMIA produced better results in terms of the NMSE, when compared to the MLP, Elman, Jordan, and the SONIA networks for nonstationary time series prediction. The Signal to Noise Ratio (SNR) compares the level of a desired signal to the level of background noise; in this case, it is the ratio of useful information of a signal compared to false or irrelevant data. The 5-step ahead predictions show consistent results. Again, the DSMIA has the best SNR for the valley maximum temperature. The results also indicated that the proposed network generated significantly better results than the SONIA networks.

It is evident from the nonstationary and stationary prediction simulation results that the transformation of the signals from nonstationary to stationary improved the results for most of the neural network architectures. For stationary prediction, the proposed DSMIA showed better results than the SONIA network for most of physical signals using the NMSE and the SNR measures. Furthermore, the proposed network show slightly improved results than all the benchmarked networks.

Table 12

Number of Hidden nodes in the proposed DSMIA and the SONIA networks for five steps ahead stationary signals using the best simulation results.

Signals	Nonstationary prediction		Stationary prediction	
	SONIA	DSMIA	SONIA	DSMIA
Valley sunshine	4	4	3	3
Valley max temp	5	5	4	3
Valley rainfall	4	4	4	3

Table 12 shows the number of hidden nodes utilized for the prediction of the physical signals on the best out of the sample simulation results between the proposed and the SONIA networks. The results indicate that the proposed network required a similar number of hidden units for the prediction of nonstationary signals. In addition, the results indicate that when the data is transformed to stationary, a smaller number of hidden units is required for both the DSMIA and the SONIA network.

To further analyse the significance of the results, we conducted a paired t-test [88] on the best simulation results to determine if there is any significant difference among the proposed DSMIA and the other neural network architectures based on the absolute value of the error. The calculated t-value showed that the proposed technique outperforms ANN with $\alpha = 5\%$ significance level for one tailed test.

6.1 DISCUSSION

In this work, several existing classification algorithms and the proposed DSMIA neural network are compared in weather data prediction. The evaluation of prediction performance has been measured using widely utilised evaluation measures for time series prediction. Two sets of experiments were conducted, for stationary and nonstationary time series prediction.

From the results obtained, the results show that the self-organized hidden layer using the immune system algorithm and dynamic links improve the predictive capabilities of the model. More importantly, the proposed DSMIA model shows promise, as the results indicate that it outperforms several neural networks. This improvement can be associated with the combination of supervised and unsupervised learning techniques used in the DSMIA model [50]. The hidden layer can cluster the input nodes to the centroids of hidden units, which gives the local network pattern of the input data. The Euclidean distance was utilized to compute the distance between the input units and the centroids of hidden units.

7. Conclusions and future work

Weather data exhibits a range of big data characteristics, for example volume, velocity, and veracity. The challenges of weather forecasting data therefore can be considered as a time series data analytics problem. In this work, the dynamic self-organized neural network inspired by the immune algorithm is proposed for the prediction of weather data signals. The nonstationary weather signals have been transformed to stationary. The main point that the dynamic self-organized multilayer neural network inspired by the immune algorithm (DSMIA) has assisted to optimise the performance due to novel combination of supervised and unsupervised learning techniques. In addition, this method performed well in data weather prediction, because it has used SOM unsupervised methods in the hidden layer and recurrent links. The simulation results showed a relative improvement achieved by the proposed network when using the average results of 30 simulations.

Since clustering methods have been widely used in various applications of data mining, changing the learning process with the adoption of unsupervised learning in the DSMIA might serve other applications, such as medical diagnostics and pattern recognition for large databases, containing many attributes. The structure of the proposed network can be adapted for clustering tasks by changing the back-propagation algorithm in the output layer, which is supervised learning algorithm to unsupervised learning algorithm.

We consider for future work the use of global optimisation algorithms such as genetic optimisation to explore more comprehensively the space of possible recurrent network architectures. We note that the current study has addressed only weather forecasting applications, which may not expose the full potential of the RNN in the classification setting. We suggest therefore that an algorithmic model search may be implemented with various application such as, flood predcation and earthquake prediction that can expand the scope and scale of this study.

References

[1] K. Krishnan, *Data warehousing in the age of big data*: Newnes, 2013.

[2] S. Sagiroglu and D. Sinanc, "Big data: A review," in *Collaboration Technologies and Systems (CTS), 2013 International Conference on*, 2013, pp. 42-47.

[3] L. Mirea and T. Marcu, "System identification using functional-link neural networks with dynamic structure," *IFAC Proceedings Volumes*, vol. 35, pp. 205-210, 2002.

[4] R. H. Chowdhury, M. B. Reaz, M. A. B. M. Ali, A. A. Bakar, K. Chellappan, and T. G. Chang, "Surface electromyography signal processing and classification techniques," *Sensors*, vol. 13, pp. 12431-12466, 2013.

[5] R. Adhikari and R. Agrawal, "An introductory study on time series modeling and forecasting," *arXiv preprint arXiv:1302.6613*, 2013.

[6] A. Phinyomark, G. Chujit, P. Phukpattaranont, C. Limsakul, and H. Hu, "A preliminary study assessing time-domain EMG features of classifying exercises in preventing falls in the elderly," in *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2012 9th International Conference on*, 2012, pp. 1-4.

[7] P. Fergus, P. Cheung, A. Hussain, D. Al-Jumeily, C. Dobbins, and S. Iram, "Prediction of preterm deliveries from EHG signals using machine learning," *PloS one*, vol. 8, p. e77154, 2013.

[8] D. Roverso, "Multivariate temporal classification by windowed wavelet decomposition and recurrent neural networks," in *3rd ANS international topical meeting on nuclear plant instrumentation, control and human-machine interface*, 2000.

[9] F. Filippetti, A. Uncini, C. Piazza, P. Campolucci, C. Tassoni, and G. Franceschini, "Neural network architectures for fault diagnosis and parameter recognition in induction machines," in *Electrotechnical Conference, 1996. MELECON'96., 8th Mediterranean*, 1996, pp. 289-293.

[10] R. Allan and W. Kinsner, "A study of microscopic images of human breast disease using competitive neural networks," in *Electrical and Computer Engineering, 2001. Canadian Conference on*, 2001, pp. 289-293.

[11] W. You, Y. Wang, B. Wo, S. Lv, A. Zhan, and W. Sun, "Recognition of Coronary Heart Disease Patients by RBF Neural Network Basing on Contents of Microelements in Human Blood," in *Computational Intelligence and Design, 2009. ISCID'09. Second International Symposium on*, 2009, pp. 409-412.

[12] L. Zhang, P. Luh, and K. Kasiviswanathan, "Energy Clearing Price Prediction and Confidence Interval Estimation With Cascaded Neural Networks," *IEEE Power Engineering Review*, vol. 22, pp. 60-60, 2002.

[13] H. Marzi and M. Turnbull, "Use of neural networks in forecasting financial market," in *Granular Computing, 2007. GRC 2007. IEEE International Conference on*, 2007, pp. 516-516.
[14] D. He and R. Liu, "Ultra-short-term wind power prediction using ANN ensemble based on PCA," in *Power Electronics and Motion Control Conference (IPEMC), 2012 7th International*, 2012, pp. 2108-2112.
[15] J. L. Herrera, "Time series prediction using inductive reasoning techniques," Universitat Politècnica de Catalunya, 2000.
[16] M. Khalaf, A. J. Hussain, D. Al-Jumeily, R. Keenan, P. Fergus, and I. O. Idowu, "Robust Approach for Medical Data Classification and Deploying Self-Care Management System for Sickle Cell Disease," in *Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM), 2015 IEEE International Conference on*, 2015, pp. 575-580.
[17] L. Cao and F. E. Tay, "Financial forecasting using support vector machines," *Neural Computing & Applications,* vol. 10, pp. 184-192, 2001.
[18] C. L. Giles, S. Lawrence, and A. C. Tsoi, "Noisy time series prediction using recurrent neural networks and grammatical inference," *Machine learning,* vol. 44, pp. 161-183, 2001.
[19] M. R. Widyanto, H. Nobuhara, K. Kawamoto, K. Hirota, and B. Kusumoputro, "Improving recognition and generalization capability of back-propagation NN using a self-organized network inspired by immune algorithm (SONIA)," *Applied Soft Computing,* vol. 6, pp. 72-84, 2005.
[20] R. Aamodt, "Using artificial neural networks to forecast financial time series," ed: Institutt for datateknikk og informasjonsvitenskap, 2010.
[21] J. Tang and X. Zhang, "Prediction of smoothed monthly mean sunspot number based on chaos theory," 2012.
[22] C. Dunis and M. Williams, "Modelling and trading the EUR/USD exchange rate: Do neural network models perform better?," *Derivatives use, trading and regulation,* vol. 8, pp. 211-239, 2002.
[23] V. A. Makarov, Y. Song, M. G. Velarde, D. Hübner, and H. Cruse, "Elements for a general memory structure: properties of recurrent neural networks used to form situation models," *Biological cybernetics,* vol. 98, pp. 371-395, 2008.
[24] S. Lohr, "The origins of 'Big Data': An etymological detective story," *New York Times,* vol. 1, 2013.
[25] S. Brunswicker, E. Bertino, and S. Matei, "Big data for open digital Innovation–A research roadmap," *Big Data Research,* vol. 2, pp. 53-58, 2015.
[26] X. Jin, B. W. Wah, X. Cheng, and Y. Wang, "Significance and challenges of big data research," *Big Data Research,* vol. 2, pp. 59-64, 2015.
[27] E. Al Nuaimi, H. Al Neyadi, N. Mohamed, and J. Al-Jaroodi, "Applications of big data to smart cities," *Journal of Internet Services and Applications,* vol. 6, p. 25, 2015.
[28] J. W. Taylor and R. Buizza, "Neural network load forecasting with weather ensemble predictions," *IEEE Transactions on Power Systems,* vol. 17, pp. 626-632, 2002.
[29] M. Khalaf, A. J. Hussain, R. Keight, D. Al-Jumeily, R. Keenan, P. Fergus*, et al.*, "The utilisation of composite machine learning models for the classification of medical datasets for sickle cell disease," in *Digital Information Processing and Communications (ICDIPC), 2016 Sixth International Conference on*, 2016, pp. 37-41.
[30] P. Zikopoulos and C. Eaton, *Understanding big data: Analytics for enterprise class hadoop and streaming data*: McGraw-Hill Osborne Media, 2011.
[31] H. Jagadish, "Big data and science: myths and reality," *Big Data Research,* vol. 2, pp. 49-52, 2015.

[32] K. A. Ismail, M. A. Majid, J. M. Zain, and N. A. A. Bakar, "Big Data prediction framework for weather Temperature based on MapReduce algorithm," in *Open Systems (ICOS), 2016 IEEE Conference on*, 2016, pp. 13-17.

[33] G. He, G. Li, X. Zou, and P. S. Ray, "Applications of a velocity Dealiasing scheme to data from the China new generation weather radar system (CINRAD)," *Weather and Forecasting,* vol. 27, pp. 218-230, 2012.

[34] A. Grillenberger and R. Romeike, "Big data–challenges for computer science education," in *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives*, 2014, pp. 29-40.

[35] P. D. Diamantoulakis, V. M. Kapinas, and G. K. Karagiannidis, "Big Data Analytics for Dynamic Energy Management in Smart Grids," *Big Data Research,* vol. 2, pp. 94-101, 2015/09/01/ 2015.

[36] Y. Qin, H. K. Yalamanchili, J. Qin, B. Yan, and J. Wang, "The Current Status and Challenges in Computational Analysis of Genomic Big Data," *Big Data Research,* vol. 2, pp. 12-18, 2015/03/01/ 2015.

[37] Y. Shi, "Big Data: history, current status, and challenges going forward," *Bridge,* vol. 44, pp. 6-11, 2014.

[38] H. Hassani and E. S. Silva, "Forecasting with big data: A review," *Annals of Data Science,* vol. 2, pp. 5-19, 2015.

[39] "ICFR 2013: Experiences in Asia and Europe,International Conference on Flood Resilience," vol. University of Exeter, UK, 5-7 September 2013.

[40] V. Nourani, Ö. Kisi, and M. Komasi, "Two hybrid artificial intelligence approaches for modeling rainfall–runoff process," *Journal of Hydrology,* vol. 402, pp. 41-59, 2011.

[41] M. Taiana, J. Nascimento, and A. Bernardino, "On the purity of training and testing data for learning: The case of pedestrian detection," *Neurocomputing,* vol. 150, pp. 214-226, 2015.

[42] M. Rocha, P. Cortez, and J. Neves, "Evolution of neural networks for classification and regression," *Neurocomputing,* vol. 70, pp. 2809-2816, 2007.

[43] A. J. Hussain, P. Fergus, H. Al-Askar, D. Al-Jumeily, and F. Jager, "Dynamic neural network architecture inspired by the immune algorithm to predict preterm deliveries in pregnant women," *Neurocomputing,* vol. 151, pp. 963-974, 2015.

[44] G. Shrivastava, S. Karmakar, M. K. Kowar, and P. Guhathakurta, "Application of artificial neural networks in weather forecasting: a comprehensive literature review," *International Journal of Computer Applications,* vol. 51, 2012.

[45] T. Chow and C. Leung, "Neural network based short-term load forecasting using weather compensation," *IEEE Transactions on Power Systems,* vol. 11, pp. 1736-1742, 1996.

[46] K. l. Hsu, H. V. Gupta, and S. Sorooshian, "Artificial neural network modeling of the rainfall-runoff process," *Water resources research,* vol. 31, pp. 2517-2530, 1995.

[47] S. S. Baboo and I. K. Shereef, "An efficient weather forecasting system using artificial neural network," *International journal of environmental science and development,* vol. 1, p. 321, 2010.

[48] D. Grimes, E. Coppola, M. Verdecchia, and G. Visconti, "A neural network approach to real-time rainfall estimation for Africa using satellite data," *Journal of Hydrometeorology,* vol. 4, pp. 1119-1133, 2003.

[49] M. A. Mandale and M. Jadhawar, "Weather forecast prediction: a Data Mining application."

[50] P. S. Dutta and H. Tahbilder, "Prediction of rainfall using data mining technique over Assam," *IJCSE,* vol. 5, pp. 85-90, 2014.

[51] J. L. Elman, "Finding structure in time," *Cognitive science,* vol. 14, pp. 179-211, 1990.

[52] M. I. Jordan, "Attractor dynamics and parallellism in a connectionist sequential machine," 1986.
[53] E. Hannan and M. Deistler, "The statistical theory of linear systemsWiley," *New York,* 1988.
[54] R. d. A. Araújo and T. A. Ferreira, "An intelligent hybrid morphological-rank-linear method for financial time series prediction," *Neurocomputing,* vol. 72, pp. 2507-2524, 2009.
[55] S. Singh, P. Bhambri, and J. Gill, "Time series based temperature prediction using back propagation with genetic algorithm technique," *International Journal of Computer Science,* vol. 8, pp. 28-35, 2011.
[56] A. B. Watkins, "AIRS: A resource limited artificial immune classifier," Mississippi State University Mississippi, 2001.
[57] R. Ghazali, A. J. Hussain, N. M. Nawi, and B. Mohamad, "Non-stationary and stationary prediction of financial time series using dynamic ridge polynomial neural network," *Neurocomputing,* vol. 72, pp. 2359-2367, 2009.
[58] S. G. Mengistu, C. G. Quick, and I. F. Creed, "Nutrient export from catchments on forested landscapes reveals complex nonstationary and stationary climate signals," *Water Resources Research,* vol. 49, pp. 3863-3880, 2013.
[59] M. A. Zaytar and C. El Amrani, "Sequence to sequence weather forecasting with long short term memory recurrent neural networks," *Int J Comput Appl,* vol. 143, 2016.
[60] R. Chandra, "Competition and collaboration in cooperative coevolution of Elman recurrent neural networks for time-series prediction," *IEEE transactions on neural networks and learning systems,* vol. 26, pp. 3123-3136, 2015.
[61] I. Maqsood, M. R. Khan, and A. Abraham, "An ensemble of neural networks for weather forecasting," *Neural Computing & Applications,* vol. 13, pp. 112-122, 2004.
[62] I. S. Isa, S. Omar, Z. Saad, N. M. Noor, and M. K. Osman, "Weather forecasting using photovoltaic system and neural network," in *Computational Intelligence, Communication Systems and Networks (CICSyN), 2010 Second International Conference on,* 2010, pp. 96-100.
[63] N. F. Güler, E. D. Übeyli, and İ. Güler, "Recurrent neural networks employing Lyapunov exponents for EEG signals classification," *Expert Systems with Applications,* vol. 29, pp. 506-514, 2005.
[64] A. Petrosian, D. Prokhorov, W. Lajara-Nanson, and R. Schiffer, "Recurrent neural network-based approach for early recognition of Alzheimer's disease in EEG," *Clinical Neurophysiology,* vol. 112, pp. 1378-1387, 2001.
[65] A. Petrosian, D. Prokhorov, R. Homan, R. Dasheiff, and D. Wunsch Ii, "Recurrent neural network based prediction of epileptic seizures in intra- and extracranial EEG," *Neurocomputing,* vol. 30, pp. 201-218, 1// 2000.
[66] F. Visin, K. Kastner, K. Cho, M. Matteucci, A. Courville, and Y. Bengio, "ReNet: A Recurrent Neural Network Based Alternative to Convolutional Networks," *arXiv preprint arXiv:1505.00393,* 2015.
[67] E. D. Übeyli, "Analysis of EEG signals by implementing eigenvector methods/recurrent neural networks," *Digital Signal Processing,* vol. 19, pp. 134-143, 2009.
[68] M. Hüsken and P. Stagge, "Recurrent neural networks for time series classification," *Neurocomputing,* vol. 50, pp. 223-235, 2003.
[69] S. Haykin and N. Network, "A comprehensive foundation," *Neural Networks,* vol. 2, 2004.
[70] M. Khalaf, A. J. Hussain, R. Keight, D. Al-Jumeily, P. Fergus, R. Keenan, *et al.*, "Machine learning approaches to the application of disease modifying therapy for sickle cell using classification models," *Neurocomputing*.

[71] J. R. Chung, J. Kwon, and Y. Choe, "Evolution of recollection and prediction in neural networks," in *Neural Networks, 2009. IJCNN 2009. International Joint Conference on*, 2009, pp. 571-577.
[72] S. Ling, F. H. Leung, K. Leung, H. Lam, and H. Iu, *An improved GA based modified dynamic neural network for Cantonese-digit speech recognition*: InTech Open Access Publisher, 2007.
[73] E. M. Forney and C. W. Anderson, "Classification of EEG during imagined mental tasks by forecasting with Elman recurrent neural networks," in *Neural Networks (IJCNN), The 2011 International Joint Conference on*, 2011, pp. 2749-2755.
[74] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735-1780, 1997.
[75] F. Gers, "Long short-term memory in recurrent neural networks," *Unpublished PhD dissertation, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland*, 2001.
[76] K. Kawakami, "Supervised Sequence Labelling with Recurrent Neural Networks," Ph. D. thesis, Technical University of Munich, 2008.
[77] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Networks*, vol. 18, pp. 602-610, 2005.
[78] A. Graves, S. Fern, #225, ndez, F. Gomez, #252, *et al.*, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," presented at the Proceedings of the 23rd international conference on Machine learning, Pittsburgh, Pennsylvania, USA, 2006.
[79] S. Hochreiter, M. Heusel, and K. Obermayer, "Fast model-based protein homology detection without alignment," *Bioinformatics*, vol. 23, pp. 1728-1736, 2007.
[80] M. Liwicki, A. Graves, S. Fernàndez, H. Bunke, and J. Schmidhuber, "A novel approach to on-line handwriting recognition based on bidirectional long short-term memory networks," in *Proceedings of the 9th International Conference on Document Analysis and Recognition, ICDAR 2007*, 2007.
[81] B. Bakker, "Reinforcement learning with long short-term memory," in *Advances in neural information processing systems*, 2002, pp. 1475-1482.
[82] D. M. Nelson, A. C. Pereira, and R. A. de Oliveira, "Stock market's price movement prediction with LSTM neural networks," in *Neural Networks (IJCNN), 2017 International Joint Conference on*, 2017, pp. 1419-1426.
[83] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," *Transportation Research Part C: Emerging Technologies*, vol. 54, pp. 187-197, 2015.
[84] O. A. Abidogun, "Data mining, fraud detection and mobile telecommunications: call pattern analysis with unsupervised neural networks," University of the Western Cape, 2005.
[85] Q. Zhang, H. Wang, J. Dong, G. Zhong, and X. Sun, "Prediction of Sea Surface Temperature using Long Short-Term Memory," *arXiv preprint arXiv:1705.06861*, 2017.
[86] A. Y. Hannun, A. L. Maas, D. Jurafsky, and A. Y. Ng, "First-pass large vocabulary continuous speech recognition using bi-directional recurrent DNNs," *arXiv preprint arXiv:1408.2873*, 2014.
[87] Z. Yu, D. S. Moirangthem, and M. Lee, "Continuous Timescale Long-Short Term Memory Neural Network for Human Intent Understanding," *Frontiers in neurorobotics*, vol. 11, p. 42, 2017.
[88] C. M. Douglas and C. R. George, "Applied statistics and probability for engineers," *New York7 John Wiley*, 1999.