We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

186,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Case Studies for Applications of Elman Recurrent Neural Networks

Elif Derya Übeyli[1] and Mustafa Übeyli[2]
[1]Department of Electrical and Electronics Engineering, Faculty of Engineering, TOBB Ekonomi ve Teknoloji Üniversitesi, Ankara,
[2]Department of Mechanical Engineering, Faculty of Engineering, TOBB Ekonomi ve Teknoloji Üniversitesi, 06530 Söğütözü, Ankara,
Turkey

1. Introduction

Artificial neural networks (ANNs) are computational modeling tools that have recently emerged and found extensive acceptance in many disciplines for modeling complex real-world problems. ANN-based models are empirical in nature, however they can provide practically accurate solutions for precisely or imprecisely formulated problems and for phenomena that are only understood through experimental data and field observations. ANNs produce complicated nonlinear models relating the inputs (the independent variables of a system) to the outputs (the dependent predictive variables). ANNs have been widely used for various tasks, such as pattern classification, time series prediction, nonlinear control, and function approximation. ANNs are desirable because (i) nonlinearity allows better fit to the data, (ii) noise-insensitivity provides accurate prediction in the presence of uncertain data and measurement errors, (iii) high parallelism implies fast processing and hardware failure-tolerance, (iv) learning and adaptivity allow the system to modify its internal structure in response to changing environment, and (v) generalization enables application of the model to unlearned data (Fausett, 1994; Haykin, 1994; Hassoun, 1995).

The idea of using ANNs for pattern classification purposes has encountered, for a long time, the favour of many researchers (Miller et al., 1992; Wright et al., 1997; Wright & Gough, 1999; Saxena et al., 2002; Übeyli, 2007a; 2007b; 2008a; 2008b; 2008c). Feedforward neural networks are a basic type of neural networks capable of approximating generic classes of functions, including continuous and integrable ones. One of the most frequently used feedforward neural network for pattern classification is the multilayer perceptron neural network (MLPNN) which is trained to produce a spatial output pattern in response to an input spatial pattern (Fausett, 1994; Haykin, 1994; Hassoun, 1995). The mapping performed is static, therefore, the network is inherently not suitable for processing temporal patterns. Attempts have been made to use the MLPNN to classify temporal patterns by transforming the temporal domain into a spatial domain.

An alternate neural network approach is to use recurrent neural networks (RNNs) which have memory to encode past history. Several forms of RNNs have been proposed and they may be classified as partially recurrent or fully recurrent networks (Saad et al., 1998; Gupta

& McAvoy, 2000; Gupta et al., 2000; Übeyli & Übeyli, 2007; Übeyli, 2008a; 2008c). RNNs can perform highly non-linear dynamic mappings and thus have temporally extended applications, whereas multilayer feedforward networks are confined to performing static mappings. RNNs have been used in a number of interesting applications including associative memories, spatiotemporal pattern classification, control, optimization, forecasting and generalization of pattern sequences (Saad et al., 1998; Gupta & McAvoy, 2000; Gupta et al., 2000; Übeyli & Übeyli, 2007; Übeyli, 2008a; 2008c). In partially recurrent networks, partial recurrence is created by feeding back delayed hidden unit outputs or the outputs of the network as additional input units. The partially recurrent networks, whose connections are mainly feedforward were used, but they include a carefully chosen set of feedback connections. One example of such a network is an Elman RNN which in principle is set up as a regular feedforward network (Elman, 1990). Architecture of Elman RNNs, case studies for biomedical engineering, case study for nuclear engineering are presented in the subtitles of this chapter. The results of the case studies for biomedical engineering and nuclear engineering are presented. These conclusions will assist to the readers in gaining intuition about the performance of the Elman RNNs used in biomedical engineering and nuclear engineering problems.

2. Architecture of Elman recurrent neural networks

RNNs have been used in pattern classification, control, optimization, forecasting and generalization of pattern sequences (Petrosian et al., 2000; Petrosian et al., 2001; Shieh et al., 2004; Übeyli & Übeyli, 2007; Übeyli, 2008a; 2008c). Fully recurrent networks use unconstrained fully interconnected architectures and learning algorithms that can deal with time-varying input and/or output in non-trivial ways. In spite of several modifications of learning algorithms to reduce the computational expense, fully recurrent networks are still complicated when dealing with complex problems. Therefore, the partially recurrent networks, whose connections are mainly feedforward, were used but they include a carefully chosen set of feedback connections. The recurrence allows the network to remember cues from the past without complicating the learning excessively. The structure proposed by Elman (1990) is an illustration of this kind of architecture. Elman RNNs were used in these applications and therefore in the following the Elman RNN is presented.

An Elman RNN is a network which in principle is set up as a regular feedforward network. This means that all neurons in one layer are connected with all neurons in the next layer. An exception is the so-called context layer which is a special case of a hidden layer. Figure 1 shows the architecture of an Elman RNN. The neurons in the context layer (context neurons) hold a copy of the output of the hidden neurons. The output of each hidden neuron is copied into a specific neuron in the context layer. The value of the context neuron is used as an extra input signal for all the neurons in the hidden layer one time step later. Therefore, the Elman network has an explicit memory of one time lag (Elman, 1990).

Similar to a regular feedforward neural network, the strength of all connections between neurons are indicated with a weight. Initially, all weight values are chosen randomly and are optimized during the stage of training. In an Elman network, the weights from the hidden layer to the context layer are set to one and are fixed because the values of the context neurons have to be copied exactly. Furthermore, the initial output weights of the context neurons are equal to half the output range of the other neurons in the network. The Elman network can be trained with gradient descent backpropagation and optimization

methods, similar to regular feedforward neural networks (Pineda, 1987). The backpropagation has some problems for many applications. The algorithm is not guaranteed to find the global minimum of the error function since gradient descent may get stuck in local minima, where it may remain indefinitely. In addition to this, long training sessions are often required in order to find an acceptable weight solution because of the well known difficulties inherent in gradient descent optimization (Haykin, 1994; Chaudhuri & Bhattacharya, 2000). Therefore, a lot of variations to improve the convergence of the backpropagation were proposed. Optimization methods such as second-order methods (conjugate gradient, quasi-Newton, Levenberg-Marquardt) have also been used for neural networks training in recent years. The Levenberg-Marquardt algorithm combines the best features of the Gauss-Newton technique and the steepest-descent algorithm, but avoids many of their limitations. In particular, it generally does not suffer from the problem of slow convergence (Battiti, 1992; Hagan & Menhaj, 1994) and can yield a good cost function compared with the other training algorithms.

2.1. Levenberg-Marquardt algorithm
Essentially, the Levenberg-Marquardt algorithm is a least-squares estimation algorithm based on the maximum neighborhood idea. Let $E(\mathbf{w})$ be an objective error function made up of m individual error terms $e_i^2(\mathbf{w})$ as follows:

$$E(\mathbf{w}) = \sum_{i=1}^{m} e_i^2(\mathbf{w}) = \left\| f(\mathbf{w}) \right\|^2, \tag{1}$$

where $e_i^2(\mathbf{w}) = \left(y_{di} - y_i \right)^2$ and y_{di} is the desired value of output neuron i, y_i is the actual output of that neuron.

It is assumed that function $f(\cdot)$ and its Jacobian J are known at point $\mathbf{w}$. The aim of the Levenberg-Marquardt algorithm is to compute the weight vector $\mathbf{w}$ such that $E(\mathbf{w})$ is minimum. Using the Levenberg-Marquardt algorithm, a new weight vector $\mathbf{w}_{k+1}$ can be obtained from the previous weight vector $\mathbf{w}_k$ as follows:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \delta \mathbf{w}_k, \tag{2}$$

where $\delta \mathbf{w}_k$ is defined as

$$\delta \mathbf{w}_k = -(J_k^T f(\mathbf{w}_k))(J_k^T J_k + \lambda \mathbf{I})^{-1}. \tag{3}$$

In equation (3), J_k is the Jacobian of f evaluated at $\mathbf{w}_k$, λ is the Marquardt parameter, $\mathbf{I}$ is the identity matrix (Battiti, 1992; Hagan & Menhaj, 1994). The Levenberg-Marquardt algorithm may be summarized as follows:

i.   compute $E(\mathbf{w}_k)$,

ii. start with a small value of λ ($\lambda = 0.01$),

iii.   solve equation (3) for $\delta\mathbf{w}_k$ and compute $E(\mathbf{w}_k + \delta\mathbf{w}_k)$,

iv.   if $E(\mathbf{w}_k + \delta\mathbf{w}_k) \geq E(\mathbf{w}_k)$, increase λ by a factor of 10 and go to (iii),

v.   if $E(\mathbf{w}_k + \delta\mathbf{w}_k) < E(\mathbf{w}_k)$,   decrease λ by   a   factor   of   10,   update
$\mathbf{w}_k : \mathbf{w}_k \leftarrow \mathbf{w}_k + \delta\mathbf{w}_k$ and go to (iii).

3. Case studies for biomedical engineering

Automated biomedical signals classification algorithms can be divided into three steps: pre-processing, feature extraction/selection, and classification. The techniques developed for automated biomedical signals classification transform the mostly qualitative diagnostic criteria into a more objective quantitative signal feature classification problem (Miller et al., 1992; Wright et al., 1997; Wright & Gough, 1999; Saxena et al., 2002; Übeyli, 2007a; 2007b; 2008a; 2008b; 2008c). For pattern processing problems to be tractable requires the conversion of patterns to features, which are condensed representations of patterns, ideally containing only salient information. Selection of the neural network inputs has two meanings: 1) which components of a pattern, or 2) which set of inputs best represent a given pattern. Different diverse feature vectors can be extracted from the biomedical signals under study by using different spectral analysis methods. The features are then used in representation and/or discrimination of the biomedical signals, i.e., wavelet coefficients and Lyapunov exponents (Miller et al., 1992; Wright et al., 1997; Wright & Gough, 1999; Saxena et al., 2002; Übeyli, 2007a; 2007b; 2008a; 2008b; 2008c). Therefore, the RNNs employing single feature vector or composite features can be implemented for automated classification of biomedical signals.

3.1 Elman recurrent neural networks for analysis of Doppler ultrasound signals
The implementation of Elman RNNs with the Lyapunov exponents for Doppler ultrasound signals classification is presented. This study is based on the consideration that Doppler ultrasound signals are chaotic signals. This consideration was tested successfully using the nonlinear dynamics tools, like the computation of Lyapunov exponents. Decision making was performed in two stages: computation of Lyapunov exponents as representative features of the Doppler ultrasound signals and classification using the RNNs trained on the extracted features (Übeyli, 2008a).
Doppler ultrasound is widely used as a noninvasive method for the assessment of blood flow in both the central and peripheral circulation. It may be used to estimate blood flow, to image regions of blood flow and to locate sites of arterial disease as well as flow characteristics and resistance of ophthalmic and internal carotid arteries (Evans et al., 1989). Doppler systems are based on the principle that ultrasound, emitted by an ultrasonic transducer, is returned partially towards the transducer by the moving targets, thereby inducing a shift in frequency proportional to the emitted frequency and the velocity along the ultrasound beam. Studies in the literature have shown that Doppler ultrasound evaluation can give reliable information on both systolic and diastolic blood velocities of arteries and is useful in screening certain hemodynamic alterations in arteries (Evans et al., 1989; Wright et al., 1997; Wright & Gough, 1999; Übeyli, 2008a).
The objective of the present study in the field of automated diagnosis of arterial diseases is to extract the representative features of the ophthalmic arterial (OA) and internal carotid

arterial (ICA) Doppler ultrasound signals and to present the accurate classification model. As in traditional pattern recognition systems, the model consists of three main modules: a feature extractor that generates a feature vector from the raw Doppler ultrasound signals, feature selection that composes composite features (Lyapunov exponents), and a feature classifier that outputs the class based on the composite features (recurrent neural networks – RNNs). A significant contribution of the present work was the composition of composite features which were used to train novel classifier (RNNs trained on computed Lyapunov exponents) for the OA and ICA Doppler ultrasound signals. To evaluate performance of the RNNs trained with the Levenberg-Marquardt algorithm, the classification accuracies and the central processing unit (CPU) times of training were considered.

The technique used in the computation of Lyapunov exponents was related with the Jacobi-based algorithms. For each OA and ICA Doppler segment (256 discrete data), 128 Lyapunov exponents were computed. The computed Lyapunov exponents samples of OA and ICA Doppler signals are shown in Figures 2 and 3. High-dimension of feature vectors increased computational complexity and therefore, in order to reduce the dimensionality of the extracted feature vectors (feature selection), statistics over the set of the Lyapunov exponents were used. The following statistical features were used in reducing the dimensionality of the extracted feature vectors representing the signals under study:

1. Maximum of the Lyapunov exponents of each Doppler ultrasound signal segment.
2. Minimum of the Lyapunov exponents of each Doppler ultrasound signal segment.
3. Mean of the Lyapunov exponents of each Doppler ultrasound signal segment.
4. Standard deviation of the Lyapunov exponents of each Doppler ultrasound signal segment.

The feature vectors were computed by the usage of the MATLAB software package. The RNNs proposed for classification of the Doppler ultrasound signals were implemented by using the MATLAB software package (MATLAB version 7.0 with neural networks toolbox). The key design decisions for the neural networks used in classification are the architecture and the training process. Different network architectures were experimented and the results of the architecture studies confirmed that for the OA Doppler signals, networks with one hidden layer consisting of 20 recurrent neurons results in higher classification accuracy. The RNNs with one hidden layer were superior to models with two hidden layers for the ICA Doppler signals. The most suitable network configuration found was 15 recurrent neurons for the hidden layer.

Classification results of the classifiers were displayed by a confusion matrix. In a confusion matrix, each cell contains the raw number of exemplars classified for the corresponding combination of desired and actual network outputs. The confusion matrices showing the classification results of the classifiers used for classification of the OA and ICA Doppler signals are given in Tables 1 and 2. From these matrices one can tell the frequency with which a Doppler signal is misclassified as another. As it is seen from Table 1, healthy subjects are most often confused with subjects suffering from OA stenosis, likewise subjects suffering from ocular Behcet disease with subjects suffering from OA stenosis. From Table 2, one can see that healthy subjects are most often confused with subjects suffering from ICA stenosis, likewise subjects suffering from ICA stenosis with subjects suffering from ICA occlusion.

The test performance of the classifiers can be determined by the computation of specificity, sensitivity and total classification accuracy. The specificity, sensitivity and total classification accuracy are defined as:

Specificity: number of true negative decisions / number of actually negative cases
Sensitivity: number of true positive decisions / number of actually positive cases
Total classification accuracy: number of correct decisions / total number of cases
A true negative decision occurs when both the classifier and the physician suggested the absence of a positive detection. A true positive decision occurs when the positive detection of the classifier coincided with a positive detection of the physician.
In order to demonstrate performance of the classifiers used for classification of the OA and ICA Doppler signals, the classification accuracies (specificity, sensitivity, total classification accuracy) on the test sets and the CPU times of training (for Pentium 4, 3.00 GHz) of the RNNs are presented in Table 3. The present research demonstrated that the Lyapunov exponents are the features which well represent the Doppler ultrasound signals and the RNNs trained on these features achieved high classification accuracies (Übeyli, 2008a).

3.2 Elman recurrent neural networks for detection of electrocardiographic changes in partial epileptic patients
The aim of this study is to evaluate the diagnostic accuracy of the RNNs with composite features (wavelet coefficients and Lyapunov exponents) on the electrocardiogram (ECG) signals. Two types of ECG beats (normal and partial epilepsy) were obtained from the MIT-BIH database (Al-Aweel et al., 1999). Decision making was performed in two stages: computing composite features which were then input into the classifiers and classification using the classifiers trained on the extracted features (Übeyli, 2008c).
Epileptic seizures are associated with several changes in autonomic functions, which may lead to cardiovascular, respiratory, gastrointestinal, cutaneous, and urinary manifestations (Leutmezer et al., 2003; Rocamora et al., 2003).Cardiovascular changes have received the most attention, because of their possible contribution to sudden unexplained death. Studies have reported the importance of monitoring the ECG signal during epileptic seizures, since the seizures can trigger high risk cardiac arrhythmias. Since seizures can occur at any time in an epileptic patient, the ECG may need to be recorded for several hours or days at a time, leading to an enormous quantity of data to be studied by physicians. To reduce the time and possibility of errors, automatic computer-based algorithms have been proposed to support or replace the diagnosis and analysis performed by the physician (Miller et al., 1992; Saxena et al., 2002; Übeyli, 2007a; 2007b; 2008c). From the hours of ECG data, these algorithms can flag the periods when the patient is having a seizure and, eventually, determine from these periods if any cardiac arrhythmias occured. This study provides a highly accurate algorithm for classifying non-arrhythmic ECG waveforms as normal or partial epileptic.
The evaluation of the classification capabilities of the Elman RNNs trained with Levenberg-Marquardt algorithm was performed on the ECG signals (normal and partial epilepsy ECG beats) from the MIT-BIH database (Al-Aweel et al., 1999). As in traditional pattern recognition systems, the model consists of three main modules: a feature extractor that generates a feature vector from the ECG signals, feature selection that composes composite features (wavelet coefficients and Lyapunov exponents), and a feature classifier that outputs the class based on the composite features. A significant contribution of the work was the composition of composite features which were used to train novel classifier (RNN trained on composite feature) for the ECG signals. To evaluate performance of the classifiers, the classification accuracies, the CPU times of training and the receiver operating characteristic (ROC) curves of the classifiers were examined (Übeyli, 2008c).

The detail wavelet coefficients at the first decomposition level of the two types of ECG beats are presented in Figures 4(a) and (b), respectively. From these figures it is obvious that the detail wavelet coefficients of the two types of ECG beats are different from each other and therefore they can serve as useful parameters in discriminating the ECG signals. A smaller number of parameters called wavelet coefficients are obtained by the wavelet transform (WT). These coefficients represent the ECG signals and therefore, they are particularly important for recognition and diagnostic purposes. The Lyapunov exponents of the two types of ECG beats are shown in Figures 5(a) and (b), respectively. One can see that the Lyapunov exponents of the two types of ECG beats differ significantly from each other so they can be used for representing the ECG signals. As it is seen from Figures 5(a) and (b), there are positive Lyapunov exponents, which confirm the chaotic nature of the ECG signals. Lyapunov exponents are a quantitative measure for distinguishing among the various types of orbits based upon their sensitive dependence on the initial conditions, and are used to determine the stability of any steady-state behavior, including chaotic solutions. The reason why chaotic systems show aperiodic dynamics is that phase space trajectories that have nearly identical initial states will separate from each other at an exponentially increasing rate captured by the so-called Lyapunov exponent.

The following statistical features were used in reducing the dimensionality of the extracted diverse feature vectors representing the ECG signals:

1. Maximum of the wavelet coefficients in each subband, maximum of the Lyapunov exponents in each beat.
2. Minimum of the wavelet coefficients in each subband, minimum of the Lyapunov exponents in each beat.
3. Mean of the wavelet coefficients in each subband, mean of the Lyapunov exponents in each beat.
4. Standard deviation of the wavelet coefficients in each subband, standard deviation of the Lyapunov exponents in each beat.

Different network architectures were tested and the architecture studies confirmed that for the ECG signals, RNN with one hidden layer consisting of 20 recurrent neurons trained on a composite feature vector results in higher classification accuracy. In order to compare performance of the different classifiers, for the same classification problem the MLPNN, which is the most commonly used feedforward neural networks was also implemented. The single hidden layered (25 hidden neurons) MLPNN was used to classify the ECG signals based on a composite feature vector.

The values of the statistical parameters (specificity, sensitivity and total classification accuracy) and the CPU times of training (for Pentium 4, 3.00 GHz) of the two classifiers are presented in Table 4. ROC plots provide a view of the whole spectrum of sensitivities and specificities because all possible sensitivity/specificity pairs for a particular test are graphed. The performance of a test can be evaluated by plotting a ROC curve for the test and therefore, ROC curves were used to describe the performance of the classifiers. A good test is one for which sensitivity rises rapidly and 1-specificity hardly increases at all until sensitivity becomes high. ROC curves which are shown in Figure 6 demonstrate the performances of the classifiers on the test files. The classification results presented in Table 4 and Figure 6 (classification accuracies, CPU times of training, ROC curves) denote that the RNN trained on composite feature vectors obtains higher accuracy than that of the MLPNN (Übeyli, 2008c).

4. Case study for nuclear engineering

Considerable interest has been developed to modeling of dynamic systems with ANNs in recent years. The basic motivation is the ability of neural networks to create data driven representations of the underlying dynamics with less reliance on accurate mathematical or physical modeling. There exist many problems for which such data-driven representations offer more advantages over more traditional modeling techniques, such as availability of fast hardware implementations, ability to cope with noisy or incomplete data and ability to very fast data generation by using ordinary digital computers (Narendra & Parthasarathy, 1990; Boroushaki et al., 2002; Choi et al., 2004; Übeyli & Übeyli, 2007).

Recently, data processing algorithms based on artificial intelligence gained popularity in nuclear technology. In particular, ANNs found their application in a wide range of problems (Uhrig & Tsoukalas, 1999), such as diagnostics (Bartlett & Uhrig, 1992; Kim et al., 1992), signal validation (Fantoni & Mazzola, 1996a; 1996b), anomalies detection (Ogha & Seki, 1991; Kozma & Nabeshima, 1995; Reifman, 1997) and core monitoring (Kozma et al., 1995). ANNs allow modeling of complex systems without requiring an explicit knowledge or formulation of the relationship existing among the variables, and they can constitute a valuable alternative to structured models or empirical correlations (Thibault & Grandjean, 1991).

4.1. Elman recurrent neural networks for neutronic parameters of a thorium fusion breeder

RNNs are capable to represent arbitrary nonlinear dynamical systems (Narendra & Parthasarathy, 1990; Boroushaki et al., 2002). Learning and generalization ability, fast real time operation and ease of implementation features have made RNNs popular in the last decade. Recent works by nuclear engineering researchers demonstrated the ability of RNNs in identification of complex nonlinear plants like nuclear reactor cores (Boroushaki et al., 2002; Adalı et al., 1997; Boroushaki et al., 2003; Şeker et al., 2003; Ortiz & Requena, 2004). Übeyli & Übeyli (2007) used the Elman RNNs for the estimation of the neutronic parameters of a thorium fusion breeder.

The inputs of the implemented nine RNNs (for three types of coolant and three outputs) are atomic densities of the components used in the investigated reactor (Übeyli & Übeyli, 2007). The outputs of the computations are the main neutronic parameters; tritium breeding ratio, energy multiplication factor and net ^{233}U production. Figure 7 shows the RNNs model used in neural computation of the main neutronic parameters.

In calculations by Scale4.3, atomic densities of the blanket zone components, thicknesses and materials of the zones and reaction cross section types are entered to the prepared inputs. Then, outputs are generated by running these inputs in a personal computer. In the outputs, the reaction cross sections with respect to neutron energy groups required to compute neutronic parameters of the reactor are derived from the library. After that, these outputs are processed with a computer code to get neutronic parameters of the reactor for an operation period of 48 months (Übeyli & Acır, 2007).

The nine RNNs proposed for computation of the main neutronic parameters (tritium breeding ratio computation, energy multiplication factor and net ^{233}U production) were implemented by using the MATLAB software package (MATLAB version 7.0 with neural networks toolbox). Different network architectures were experimented and the results of the architecture studies confirmed that, networks with one hidden layer results in higher computation accuracy. The Scale 4.3 was used to generate data (Übeyli & Acır, 2007). For

neural computation of the tritium breeding ratio, energy multiplication factor and net [233]U production 49 data, consisting of input parameters and the corresponding computed values of the tritium breeding ratio, energy multiplication factor and net [233]U production, were generated for each RNN.

The test results of the RNNs implemented for three types of coolant are compared with the results of Scale 4.3 in Figures 8-10 for the tritium breeding ratio (TBR) computation, the energy multiplication factor (M) and the net [233]U production, respectively. It can be clearly seen from these Figures that the results of the RNNs presented in this study are in very good agreement with the results of Scale 4.3. The difference between the output of the network and the desired output (computed using Scale 4.3) is referred to as the error and can be measured in different ways. In this study, mean square error (MSE), mean absolute error (MAE), and correlation coefficient (r) were used for the measuring error of the RNNs during test process. The correlation coefficient is limited with the range [-1,1]. When $r = 1$ there is a perfect positive linear correlation between network output and desired output, which means that they vary by the same amount. When $r = -1$ there is a perfectly linear negative correlation between network output and desired output, that means they vary in opposite ways. When $r = 0$ there is no correlation between network output and desired output. Intermediate values describe partial correlations. In Table 5, performance evaluation parameters of the RNNs implemented for three types of coolant are given for the tritium breeding ratio computation, the energy multiplication factor and the net [233]U production during test process. The values of performance evaluation parameters and the very good agreement between the results of the RNNs and the results of Scale 4.3 support the validity of the RNNs trained with the Levenberg-Marquardt algorithm presented in this study (Übeyli & Übeyli, 2007).

5. Conclusions

ANNs may offer a potentially superior method of biomedical signal analysis to the spectral analysis methods. In contrast to the conventional spectral analysis methods, ANNs not only model the signal, but also make a decision as to the class of signal. Another advantage of ANN analysis over existing methods of biomedical signal analysis is that, after an ANN has trained satisfactorily and the values of the weights and biases have been stored, testing and subsequent implementation is rapid. The proposed combined Lyapunov exponents/RNN approach can be evaluated in discrimination of other Doppler ultrasound signals or time-varying biomedical signals. Preprocessing, feature extraction methods and ANN architectures are the main modules of an automated diagnostic systems and therefore they play important roles in determining the classification accuracies. Thus, further work can be performed for improving the classification accuracies by the usage of different preprocessing (different filtering methods), feature extraction methods (different spectral analysis methods) and ANN architectures (self-organizing map, radial basis function, mixture of experts, etc.) (Übeyli, 2008a).

The research demonstrated that the wavelet coefficients and the Lyapunov exponents are the features which well represent the ECG signals and the RNN trained on these features achieved high classification accuracies. The overall results of the RNN were better when they were trained on the computed composite features for each ECG beat. The results demonstrated that significant improvement can be achieved in accuracy by using the RNNs compared to the feedforward neural network models (MLPNNs). This may be attributed to several factors including the training algorithms, estimation of the network parameters and the scattered and mixed nature of the features. The results of the present study

demonstrated that the RNN can be used in classification of the ECG beats by taking into consideration the misclassification rates (Übeyli, 2008c).

ANNs have recently been introduced to the nuclear engineering applications as a fast and flexible vehicle to modeling, simulation and optimization. A new approach based on RNNs was presented for the neutronic parameters of a thorium fusion breeder. The results of the RNNs implemented for the tritium breeding ratio computation, energy multiplication factor and net ^{233}U production in a thorium fusion breeder and the results available in the literature obtained by using Scale 4.3 were compared. The drawn conclusions confirmed that the proposed RNNs could provide an accurate computation of the tritium breeding ratio computation, the energy multiplication factor and the net ^{233}U production of the thorium fusion breeder (Übeyli & Übeyli, 2007).

6. References

Adalı, T., Bakal, B., Sönmez, M.K., Fakory, R. & Tsaoi, C.O. (1997). Modeling nuclear reactor core dynamics with recurrent neural networks. *Neurocomputing*, Vol. 15, 363-381.

Al-Aweel, I.C., Krishnamurthy, K.B., Hausdorff, J.M., Mietus, J.E., Ives, J.R., Blum, A.S., Schomer, D.L. & Goldberger, A.L. (1999). Postictal heart rate oscillations in partial epilepsy. *Neurology*, Vol. 53, No. 7, 1590-1592.

Bartlett, E.B. & Uhrig, R.E. (1992). Nuclear power plant status diagnostics using artificial neural networks. *Nuclear Technology*, Vol. 97, 272-281.

Battiti, R. (1992). First- and second-order methods for learning: between steepest descent and Newton's method. *Neural Computation*, Vol. 4, 141-166.

Boroushaki, M., Ghofrani, M.B. & Lucas, C. (2002). Identification of a nuclear reactor core (VVER) using recurrent neural networks. *Annals of Nuclear Energy*, Vol. 29, 1225-1240.

Boroushaki, M., Ghofrani, M.B., Lucas, C. & Yazdanpanah, M.J. (2003). An intelligent nuclear reactor core controller for load following operations, using recurrent neural networks and fuzzy systems. *Annals of Nuclear Energy*, Vol. 30, 63-80.

Chaudhuri, B.B. & Bhattacharya, U. (2000). Efficient training and improved performance of multilayer perceptron in pattern classification. *Neurocomputing*, Vol. 34, 11-27.

Choi, Y.J., Kim, H.K., Baek, W.P., Chang, S.H. (2004). Hybrid accident simulation methodology using artificial neural networks for nuclear power plants. *Information Sciences*, Vol. 160, 207-224.

Elman, J.L. (1990). Finding structure in time. *Cognitive Science*, Vol. 14, No. 2, 179-211.

Evans, D.H., McDicken, W.N., Skidmore, R. & Woodcock, J.P. (1989). *Doppler Ultrasound: Physics, Instrumentation and Clinical Applications*, Wiley, Chichester.

Fantoni, P.F. & Mazzola, A. (1996a). Multiple-failure signal validation in nuclear power plants using artificial neural networks. *Nuclear Technology*, Vol. 113, 368-374.

Fantoni, P.F. & Mazzola, A. (1996b). A pattern recognition-artificial neural networks based model for signal validation in nuclear power plants. *Annals of Nuclear Energy*, Vol. 23, No. 13, 1069-1076.

Fausett, L. (1994). *Fundamentals of Neural Networks Architectures, Algorithms, and Applications*, Prentice Hall, Inc., Englewood Cliffs, NJ.

Gupta, L. & McAvoy, M. (2000). Investigating the prediction capabilities of the simple recurrent neural network on real temporal sequences. *Pattern Recognition*, Vol. 33, No. 12, 2075-2081.

Gupta, L., McAvoy, M. & Phegley, J. (2000). Classification of temporal sequences via prediction using the simple recurrent neural network. *Pattern Recognition*, Vol. 33, No. 10, 1759-1770.

Hagan, M.T. & Menhaj, M.B. (1994). Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks*, Vol. 5, No. 6, 989-993.

Hassoun, M.H. (1995). *Fundamentals of Artificial Neural Networks*, Massachusetts Institute of Technology Press, London.

Haykin, S. (1994). *Neural networks: A Comprehensive Foundation*, Macmillan, New York.

Kim, K., Aljundi, T.L. & Bartlett, E.B. (1992). Confirmation of artificial neural networks: nuclear power plant fault diagnostics. *Transactions of the American Nuclear Society*, Vol. 66, 112.

Kozma, R. & Nabeshima, K. (1995). Studies on the detection of incipient coolant boiling in nuclear reactors using artificial neural networks. *Annals of Nuclear Energy*, Vol. 22, No. 7, 483-496.

Kozma, R., Sato, S., Sakuma, M., Kitamura, M. & Sugiyama, T. (1995). Generalization of knowledge acquired by a reactor core monitoring system based on a neuro-fuzzy algorithm. *Progress in Nuclear Energy*, Vol. 29, No. 3-4, 203-214.

Leutmezer, F., Schernthaner, C., Lurger, S., Pötzelberger, K. & Baumgartner, C. (2003). Electrocardiographic changes at the onset of epileptic seizures. *Epilepsia*, Vol. 44, No. 3, 348-354.

Miller, A.S., Blott, B.H., & Hames, T.K. (1992). Review of neural network applications in medical imaging and signal processing. *Medical & Biological Engineering & Computing*, Vol. 30, 449-464.

Narendra, K.S. & Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, Vol. 1, No. 1, 4-27.

Ogha, Y. & Seki, H. (1991). Using a neural network for abnormal event identification in BWRs. *Transactions of the American Nuclear Society*, Vol. 63, 110-111.

Ortiz, J.J. & Requena, I. (2004). Using a multi-state recurrent neural network to optimize loading patterns in BWRs. *Annals of Nuclear Energy*, Vol. 31, 789-803.

Petrosian, A., Prokhorov, D., Homan, R., Dasheiff, R. & Wunsch II, D. (2000). Recurrent neural network based prediction of epileptic seizures in intra- and extracranial EEG. *Neurocomputing*, Vol. 30, 201-218.

Petrosian, A.A., Prokhorov, D.V., Lajara-Nanson, W. & Schiffer, R.B. (2001). Recurrent neural network-based approach for early recognition of Alzheimer's disease in EEG. *Clinical Neurophysiology*, Vol. 112, No. 8, 1378-1387.

Pineda, F.J. (1987). Generalization of back-propagation to recurrent neural networks. *Physical Review Letters*, Vol. 59, No. 19, 2229-2232.

Reifman, J. (1997). Survey of artificial intelligence methods for detection and identification of component faults in nuclear power plants. *Nuclear Technology*, Vol. 119, 76-97.

Rocamora, R., Kurthen, M., Lickfett, L., von Oertzen, J. & Elger, C.E. (2003). Cardiac asystole in epilepsy: Clinical and neurophysiologic features. *Epilepsia*, Vol. 44, No. 2, 179-185.

Saad, E.W., Prokhorov, D.V. & Wunsch II, D.C. (1998). Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks. *IEEE Transactions on Neural Networks*, Vol. 9, No. 6, 1456-1470.

Saxena, S.C., Kumar, V. & Hamde, S.T. (2002). Feature extraction from ECG signals using wavelet transforms for disease diagnostics. *International Journal of Systems Science*, Vol. 33, No. 13, 1073-1085.

Shieh, J-S., Chou, C-F., Huang, S-J. & Kao, M-C. (2004). Intracranial pressure model in intensive care unit using a simple recurrent neural network through time. *Neurocomputing*, Vol. 57, 239-256.

Şeker, S., Ayaz, E. & Türkcan, E. (2003). Elman's recurrent neural network applications to condition monitoring in nuclear power plant and rotating machinery. *Engineering Applications of Artificial Intelligence*, Vol. 16, 647-656.

Thibault, J. & Grandjean, B.P.A. (1991). A neural network methodology for heat transfer data analysis. *International Journal of Heat Mass Transfer*, Vol. 34, No. 8, 2063-2070.

Uhrig, R.E. & Tsoukalas, L.H. (1999). Soft computing technologies in nuclear engineering applications. *Progress in Nuclear Energy*, Vol. 34, No. 1, 13-75.

Übeyli, M. & Acır, A. (2007). Utilization of thorium in a high power density hybrid reactor with innovative coolants. *Energy Conversion and Management*, Vol. 48, 576-582.

Übeyli, E.D. & Übeyli, M. (2007). Investigating neutronic parameters of a Thorium fusion breeder with recurrent neural networks. *Journal of Fusion Energy*, Vol. 26, No. 4, 323-330.

Übeyli, E.D. (2007a). Comparison of different classification algorithms in clinical decision-making. *Expert Systems*, Vol. 24, No. 1, 17-31.

Übeyli, E.D. (2007b). ECG beats classification using multiclass support vector machines with error correcting output codes. *Digital Signal Processing*, Vol.17, No. 3, 675-684.

Übeyli, E.D. (2008a). Recurrent neural networks employing Lyapunov exponents for analysis of Doppler ultrasound signals. *Expert Systems with Applications*, Vol. 34, No. 4, 2538-2544.

Übeyli, E.D. (2008b). Wavelet/mixture of experts network structure for EEG signals classification. *Expert Systems with Applications*, Vol. 34, No. 3, 1954-1962.

Übeyli, E.D. (2008c). Recurrent neural networks with composite features for detection of electrocardiographic changes in partial epileptic patients. *Computers in Biology and Medicine*, Vol. 38, No. 3, 401-410.

Wright, I.A., Gough, N.A.J., Rakebrandt, F., Wahab, M. & Woodcock, J.P. (1997). Neural network analysis of Doppler ultrasound blood flow signals: A pilot study. *Ultrasound in Medicine & Biology*, Vol. 23, 683-690.

Wright, I.A., & Gough, N.A.J. (1999). Artificial neural network analysis of common femoral artery Doppler shift signals: Classification of proximal disease. *Ultrasound in Medicine & Biology*, Vol. 24, 735-743.
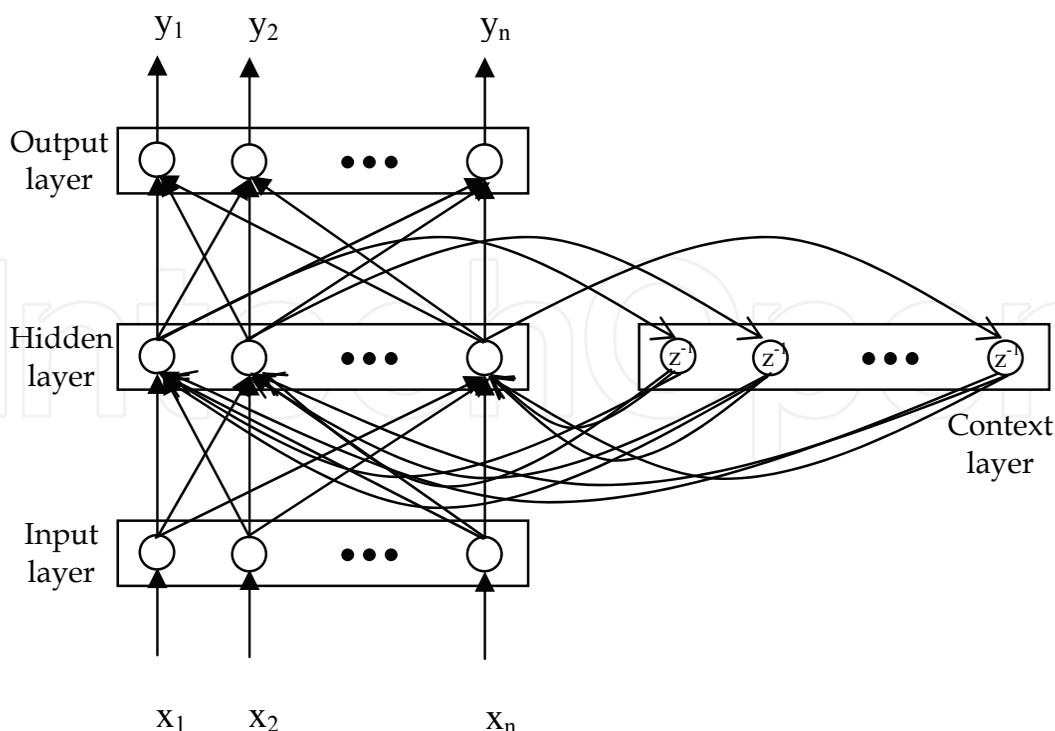
Figure 1. A schematic representation of an Elman recurrent neural network. z^{-1} represents a one time step delay unit.
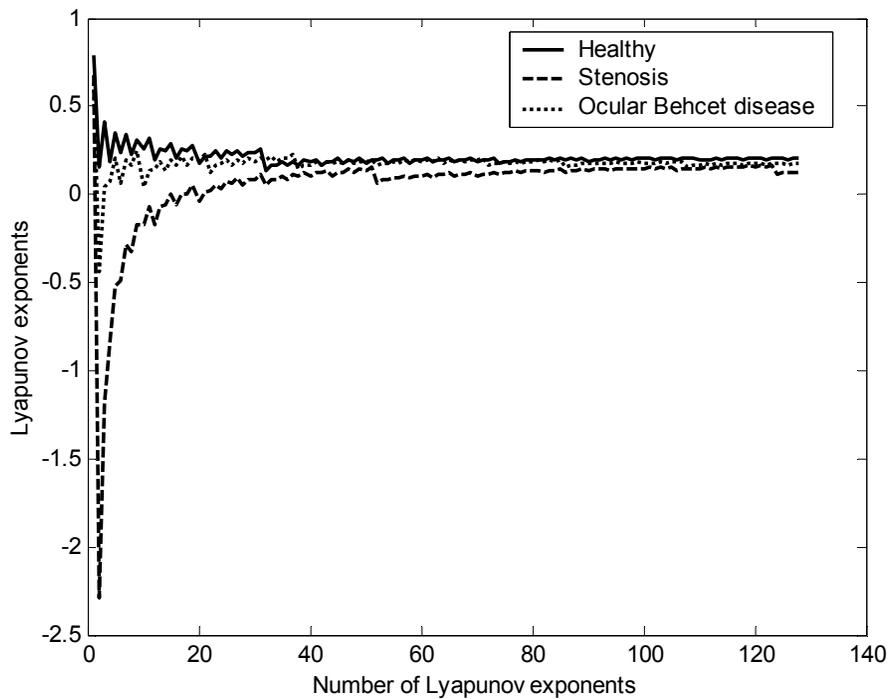
Figure 2. Lyapunov exponents of the OA Doppler signals: healthy subject (subject no: 12); subject suffering from OA stenosis (subject no: 27); subject suffering from ocular Behcet disease (subject no: 38)
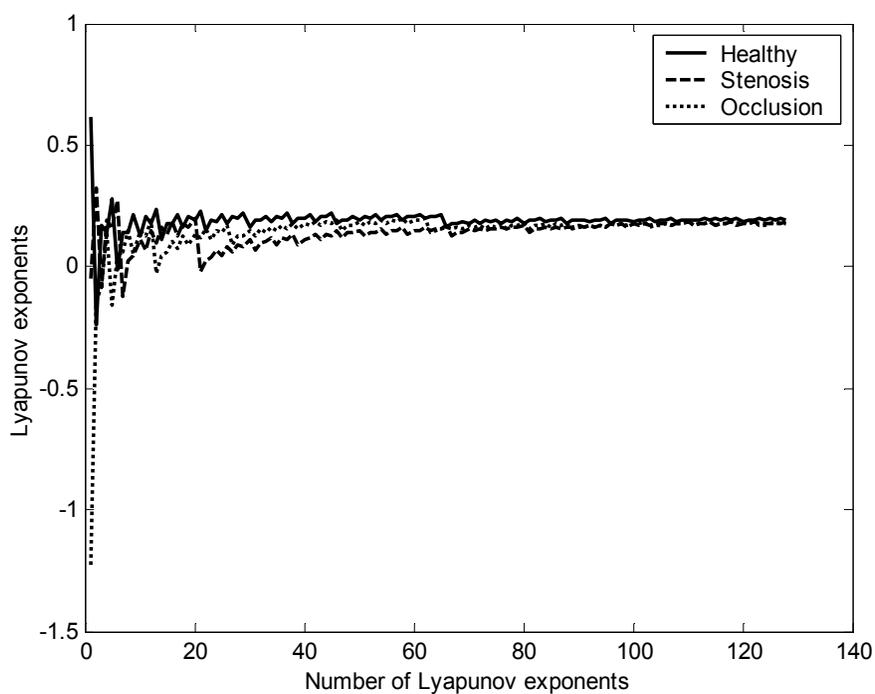

Figure 3. Lyapunov exponents of the ICA Doppler signals obtained from: a healthy subject (subject no: 15); a subject suffering from ICA stenosis (subject no: 32); a subject suffering from ICA occlusion (subject no: 43)
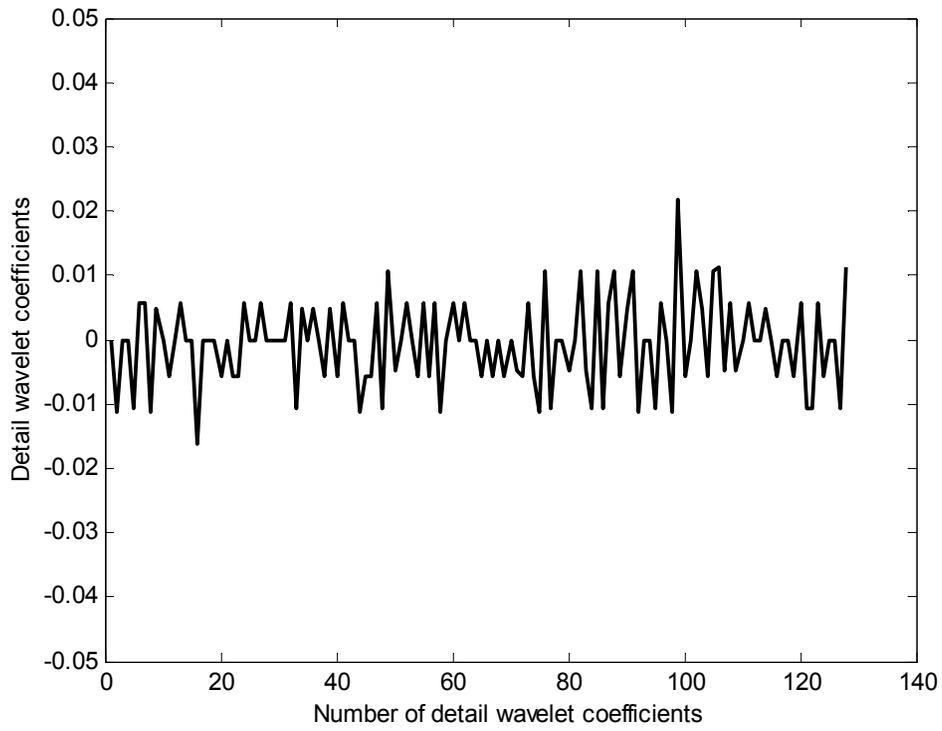
(a)
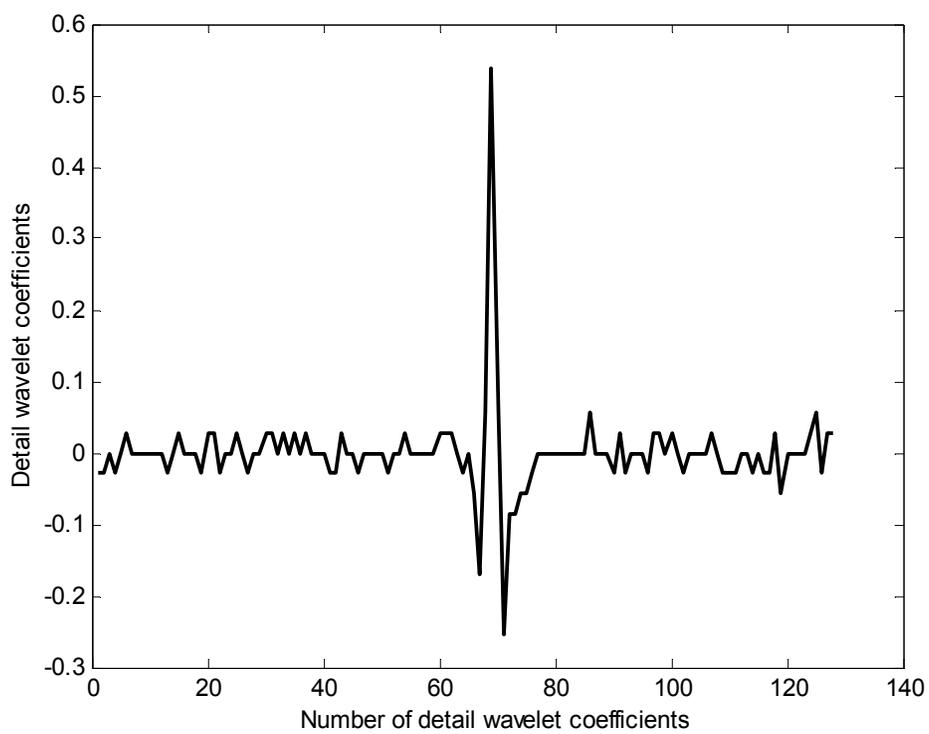
(b)

Figure 4. The detail wavelet coefficients at the first decomposition level of the ECG beats: (a) normal beat, (b) partial epilepsy beat
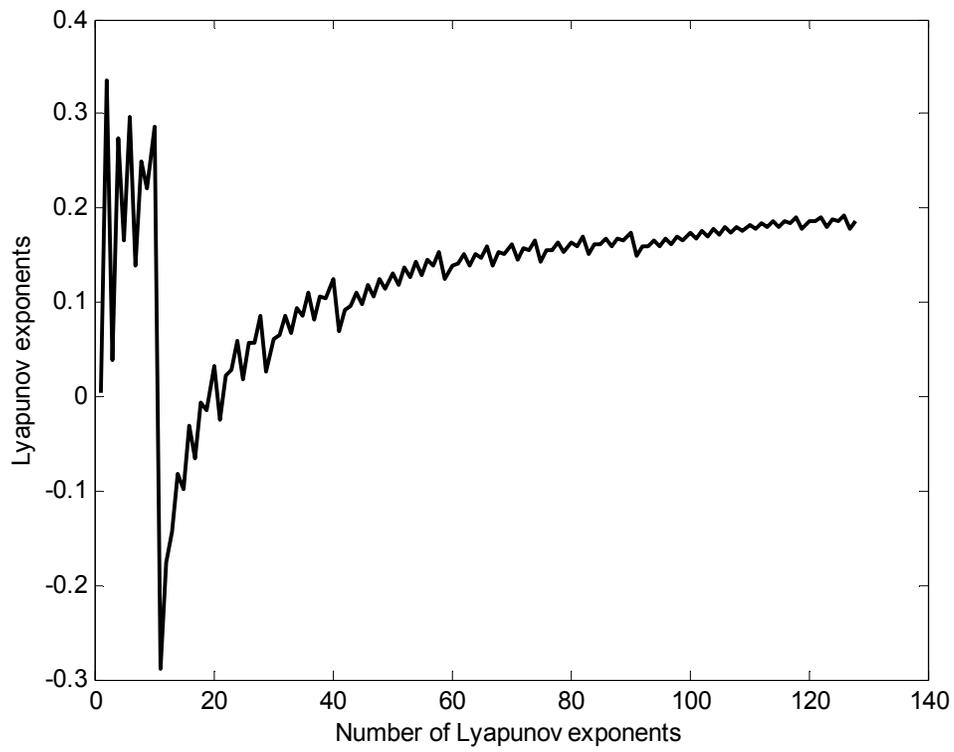
(a)
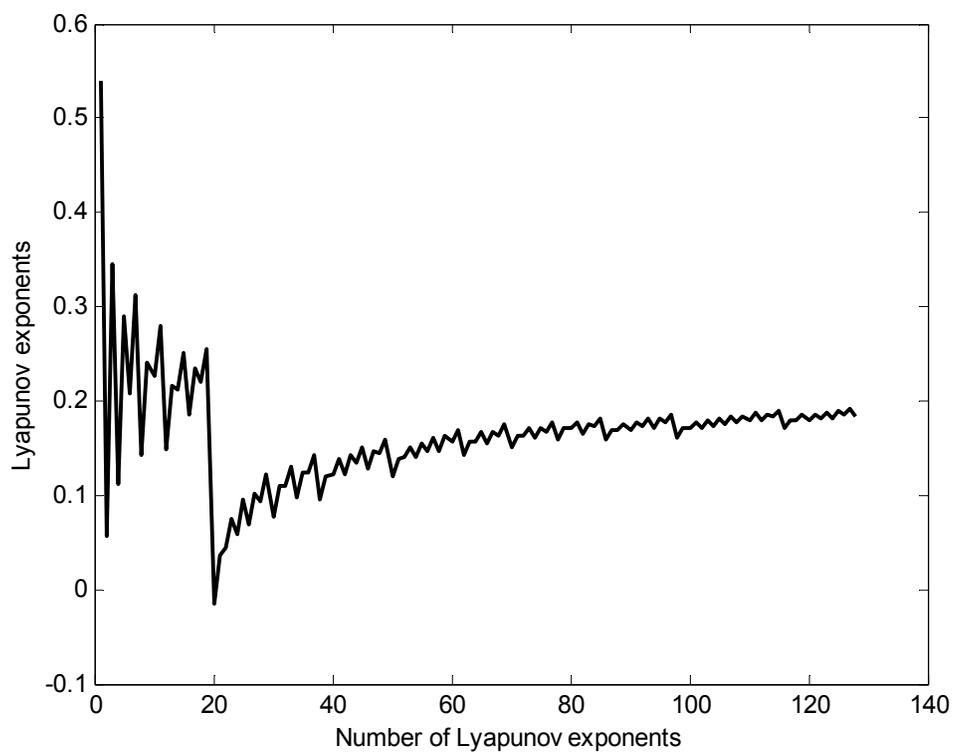
(b)

Figure 5. Lyapunov exponents of the ECG beats: (a) normal beat, (b) partial epilepsy beat
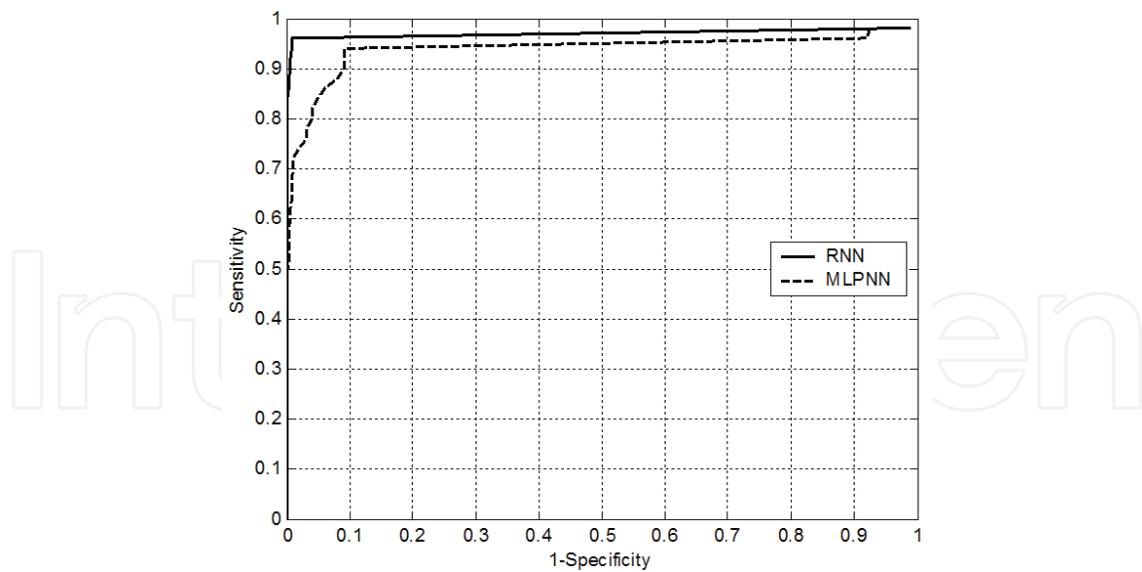
Figure 6. ROC curves of the classifiers used for classification of the ECG beats
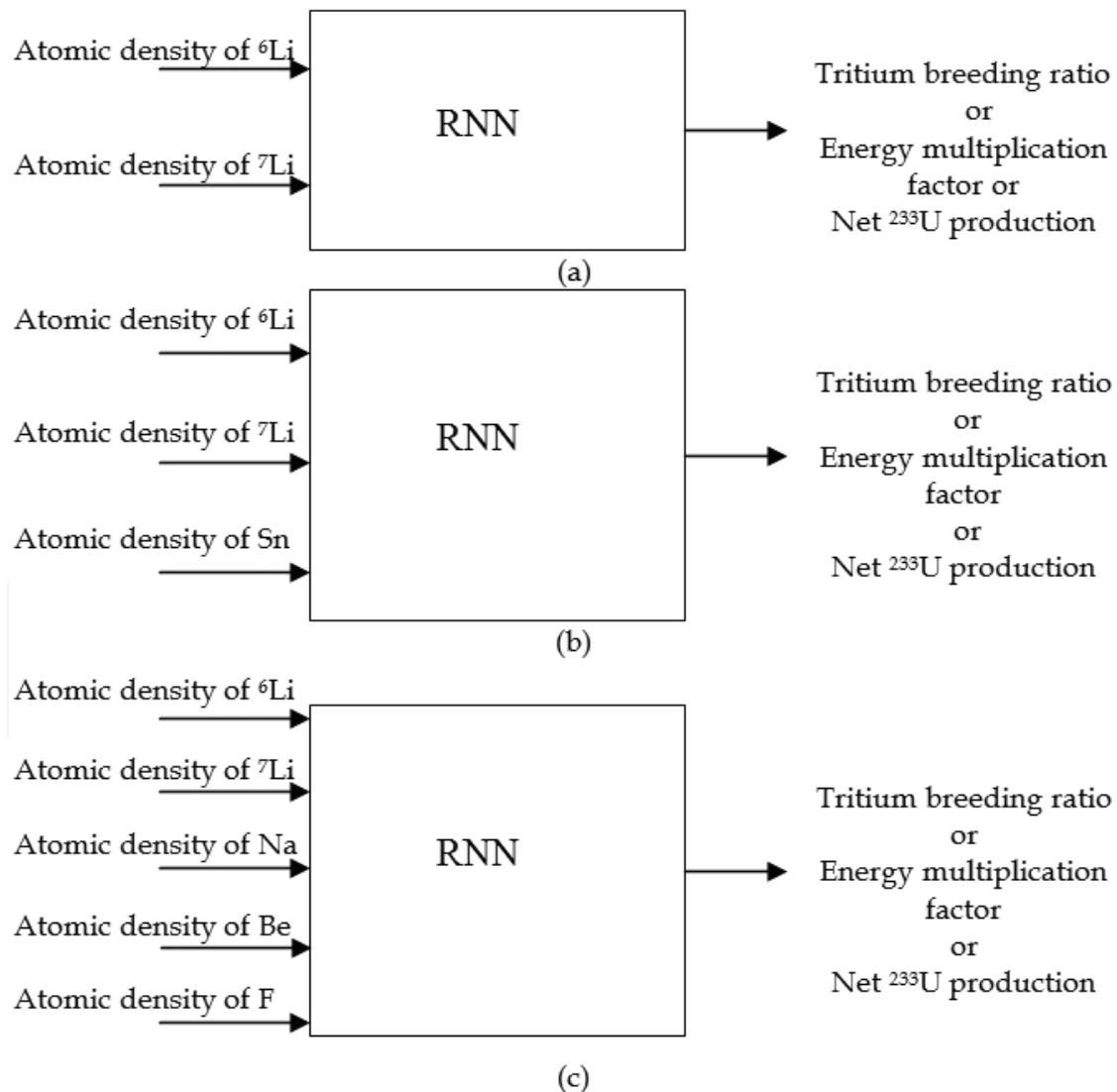
Figure 7. Implemented RNNs for various coolant type (a) Natural Lithium (b) $Li_{20}S_{80}$ (c) Flinabe
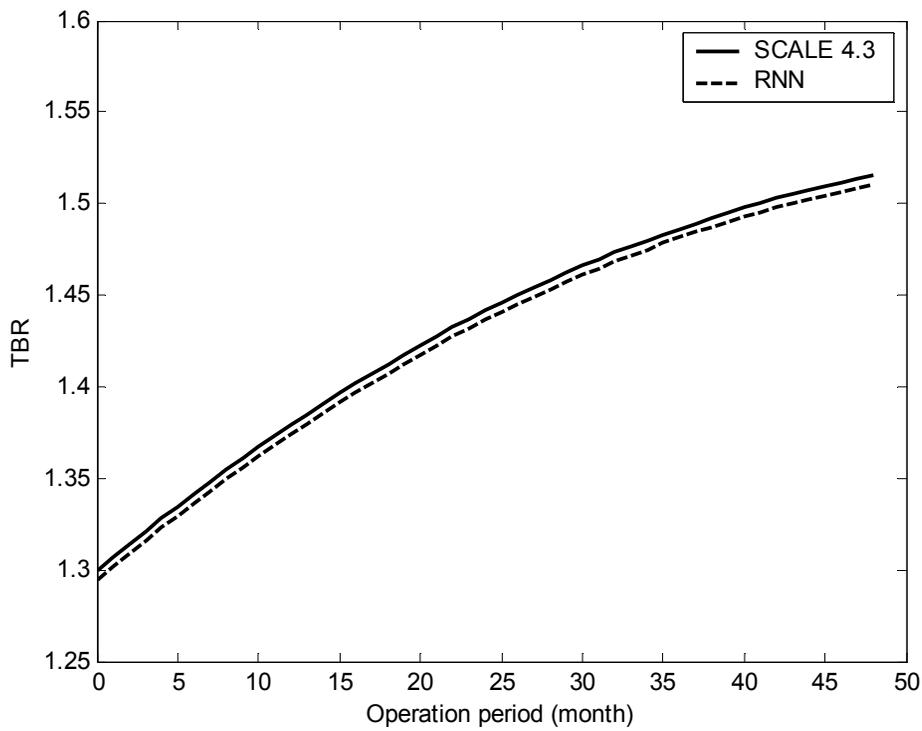
Figure 8. TBR variation in the blanket cooled with natural Li obtained by Scale 4.3 (Übeyli & Acır, 2007) and RNN
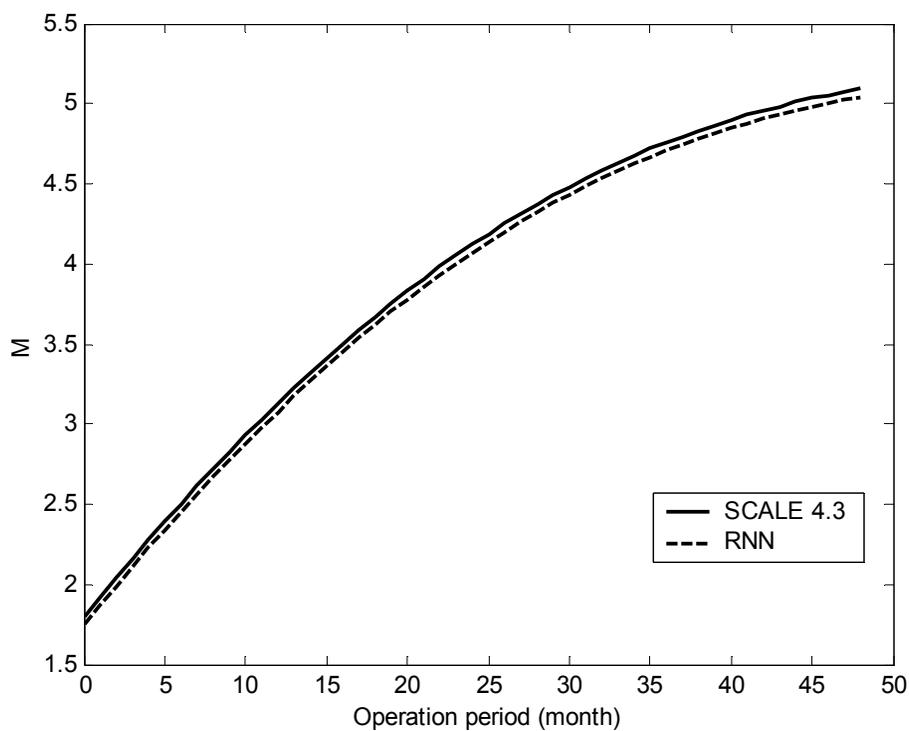
Figure 9. Change in M with respect to time in the blanket using $Li_{20}Sn_{80}$ obtained by Scale 4.3 (Übeyli & Acır, 2007) and RNN
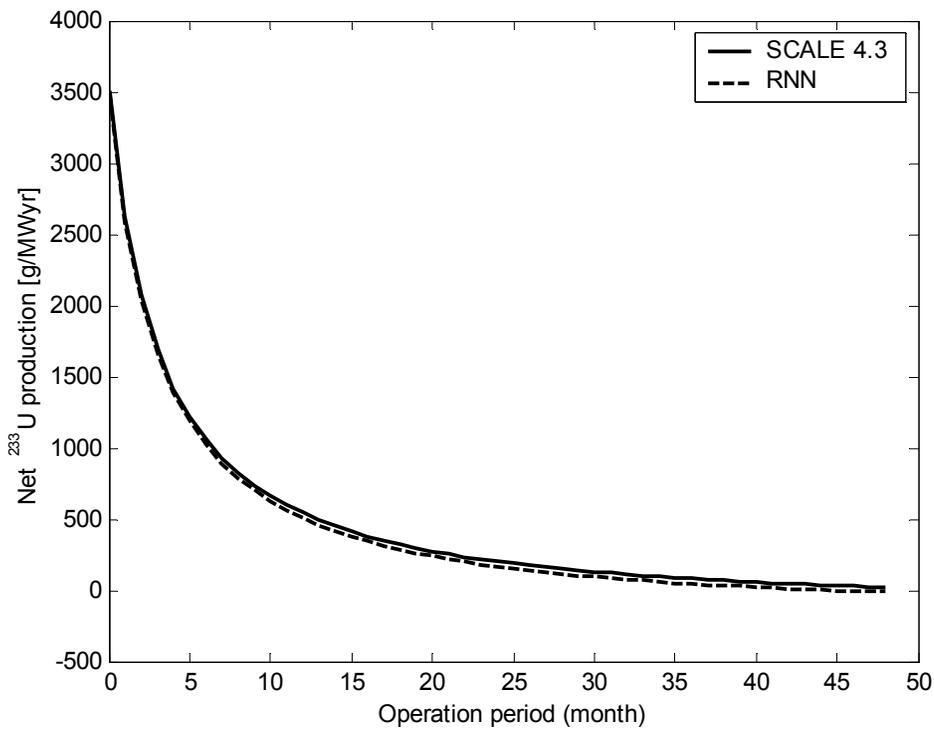
Figure 10. Net ^{233}U production in the blanket using Flinabe obtained by Scale 4.3 (Übeyli & Acır, 2007) and RNN

Desired Result	Output Result		
	Healthy	OA stenosis	Ocular Behcet disease
Healthy	41	0	0
OA stenosis	2	32	1
Ocular Behcet disease	0	0	33

Table 1. Confusion matrix of the RNN used for classification of the OA Doppler signals

Desired Result	Output Result		
	Healthy	ICA stenosis	ICA occlusion
Healthy	31	0	0
ICA stenosis	1	40	0
ICA occlusion	0	2	36

Table 2. Confusion matrix of the RNN used for classification of the ICA Doppler signals

Classifiers	CPU time (min:s)	Classification Accuracies	Values (%)
RNN implemented for OA Doppler signals	8:23	Specificity	95.35
		Sensitivity (OA stenosis)	100.00
		Sensitivity (Ocular Behcet disease)	97.06
		Total classification accuracy	97.25
RNN implemented for ICA Doppler signals	7:41	Specificity	96.88
		Sensitivity (ICA stenosis)	95.24
		Sensitivity (ICA occlusion)	100.00
		Total classification accuracy	97.27

Table 3. The classification accuracies and the CPU times of training of the classifiers used for classification of the OA and ICA Doppler signals

Classifiers	Statistical Parameters (%)			CPU time (min:s)
	Specificity	Sensitivity	Total classification accuracy	
RNN	98.89	97.78	98.33	12:34
MLPNN	92.22	93.33	92.78	17:05

Table 4. The values of the statistical parameters and the CPU times of training of the classifiers used for classification of the ECG beats

Coolant type	Performance	RNNs for tritium breeding ratio	RNNs for energy multiplication factor	RNNs for net ^{233}U production
Natural Lithium	MSE	0.009	0.005	0.008
	MAE	0.008	0.004	0.008
	r	0.892	0.921	0.945
$Li_{20}S_{80}$	MSE	0.089	0.006	0.007
	MAE	0.008	0.005	0.008
	r	0.899	0.934	0.905
Flinabe	MSE	0.009	0.005	0.006
	MAE	0.088	0.005	0.007
	r	0.895	0.924	0.936

Table 5. Performance evaluation parameters of the RNNs implemented for the estimation of the neutronic parameters of a thorium fusion breeder
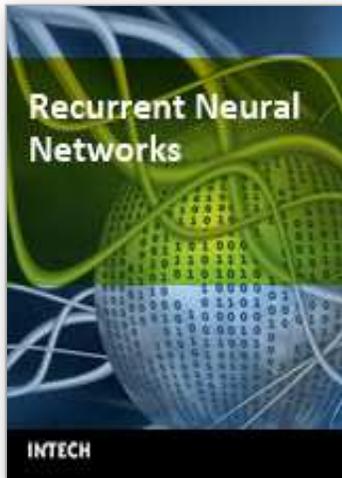
Recurrent Neural Networks

Edited by Xiaolin Hu and P. Balasubramaniam

The concept of neural network originated from neuroscience, and one of its primitive aims is to help us understand the principle of the central nerve system and related behaviors through mathematical modeling. The first part of the book is a collection of three contributions dedicated to this aim. The second part of the book consists of seven chapters, all of which are about system identification and control. The third part of the book is composed of Chapter 11 and Chapter 12, where two interesting RNNs are discussed, respectively.The fourth part of the book comprises four chapters focusing on optimization problems. Doing optimization in a way like the central nerve systems of advanced animals including humans is promising from some viewpoints.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Elif Derya Übeyli and Mustafa Übeyli (2008). Case Studies for Applications of Elman Recurrent Neural Networks, Recurrent Neural Networks, Xiaolin Hu and P. Balasubramaniam (Ed.), ISBN: 978-953-7619-08-4, InTech, Available from:
http://www.intechopen.com/books/recurrent_neural_networks/case_studies_for_applications_of_elman_recurrent_neural_networks

INTECH

open science | open minds