Sirte University Scientific Journal (Applied Sciences) *Vol.10 (1), June 2020*

Performance Evaluation of Feed Forward Neural and Recurrent Neural On Real System Dataset of Robot Execution

Ali.k. Diryag[1], Nasar A. Ali[2], kaled M. Legweel[3]
ali.k.diryag@su.edu.ly

[1,2]Department of Mechanical Engineering, Faculty of Engineering, Sirte University, Libya
[3]Faculty of Engineering science & Technology, Sabha, Libya.

Abstract

This article presents approach based on the artificial neural networks (ANN). It is employed to evaluate of performance real date set of real system. The training and testing dataset used in the experiment consists of forces and torques memorized immediately after the real robot failed in assignment execution. Two types of neural networks (NN) are utilized in order to find best performance method - feed forward neural networks (FFNN) and recurrent neural networks (RNN) and an additional evaluation would be to run test sets for each neural network to see how small an error is produced. Moreover, we investigated 24 neural structures implemented in Matlab software. The obtained results confirm that this approach can be successfully applied in this domain.

Keywords: *Artificial Neural networks, Recurent nural, Feedforward nural, real system.NN strectuers, Proformance,*

1. Introduction

Artificial Neural networks (ANN) [1] is one of the five main computational intelligence paradigms [2], and are a well-known tool used as a solution for various problems in almost all areas of engineering . They can understand the mapping between data during the training process using different learning algorithms. They can automatically define classification schemes. Artificial neural networks have great capabilities to generalize, cluster or organize data, deal with uncertainties, noisy data, and non-linear relationships [3].
The researchers of artificial neural networks presented the model of multilayer Perceptron network to resolve these problems [4]. The first and most popular network is the feed forward network; the second is the recurnt network. ANN have been nonlinear systems. Hidden patterns, which could be

independent of any mathematical models, can be found from the training data sets. If the same or similar patterns are met, ANN come up with a result with minimum mean square error (MSE). For this purpose, Matlab is chosen as an experiment environment to perform the required computations and visualizations.
To describe a neural network's structures adequately, it is necessary to specify how many layers it has, each layer's transfer function, the number of neurons in each of them and to characterize how layers are interrelated [5]. The application of neural networks in the science and engineering research has recently garnered attention in the literature [6,7, 8].
The objective and main aim of the paper is, therefore, to develop a neural network model ,To be able to evaluate the ability of FFNN and RNN Artificial Neural Networks. To examine the best structure of neural networks (for a given data set) . In this study, FFNN and RNN structures have been employed, and also presents performance comparison of different structures.

2. Theoretical Framework

1.1 Feed Forward Neural Network (FFNN)

(FFNN) allow signals to travel one way only, from input to output .there is no feedback (loop) .the output of any layer does not affect that same layer. They are extensively used in pattern recognition.

2.1.1 Multilayer Feed Forward Networks (MLFF)

The MLFF network is a member of the feed forward network architecture, and is the simplest of the networks under investigation.(MLFF) has a layered structure with more layers of nodes called hidden layers between the inputs layers and the output layers In this network, there are layers, each composed of neurons.. The input layer with a linear activation functionis fed the input values which are then multiplied by an input weight matrix, passed through the hidden layer (using the sigmoid activation function), multiplied by an output weight matrix, and finally fed to the output layer which uses a linear activation function.

2.2 Recurrent Neural Network (RNN)

The RNN is similar to the MLP in general structure except that it contains a feedback loop with unit delay from some later stage of the network back to the input layer. In this paper, Elman network is used which takes the output from the hidden layer.
There are numbers of special cases of RNN such as, Elman [9], Jordan [10] and Hopfield [11]...etc. In this paper, the Elman RNN will apply for neural structures to test in software environment so as to find optimal solution

2.2.1 Elman Recurrent Neural Network (Elman RNN)

The Elman network is one type of the partial recurrent neural networks, it was introduced by Elman [9]. The Elman RNN in this study due to the hidden layer being wider than the output layer. This wider layer allows more values to be fed back to the input, thus allowing more information to be available to the network.

3. Data and Methodology

3.1 Description of Data Sets

The data used in this paper is obtained from a real system. This data is available via well-known machine learning repository [12], and refers to the evolution of forces (F) and torques (T) during execution of a specific task. In order to correctly evaluate and compare various structures, the robor excution failures in approach to grasp position are considered. Each feature in the dataset represents a force or torque value measured immediately after failure detection. Total number of instances is 88, and each instance consists of 15 sensor measurements (i.e. samples) collected at regular time intervals. Three values of forces and torques are founded in each sample; therefore, one instance has 90 different features (i.e. the values of (F) and(T).

In the dataset, 4 different robot situations (i.e. data classes) can be identified: normal, collision, obstruction and front collision with the distribution of 24 %, 19 %,, 18 %, and 39 %,, respectively. The identification of particular class is based on the values and relationships between measured forces and torques.

3.2 Artificial Neural Network (ANN) Modelling.

In this study 24 , different structures were investigated, including the networks with one, two or three hidden layers. The feedfowrd network structure in Figure (1) as 3-2 means that there are 3 neurons in the first hidden layer, 2 in the second hidden layer. The NN input and output are single column vectors since they represent scaled values of recorded measurements and corresponding to situations.

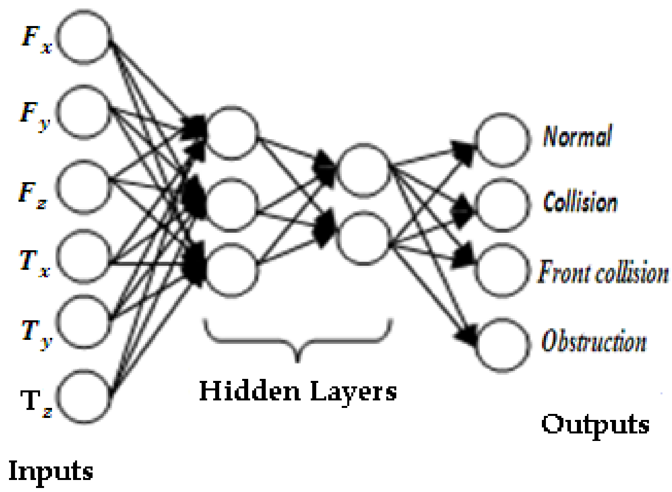

Figure 1. Feedfowrd (NN) Structure

In this paper, according to the above strategy for varying of different network structures and based on the program package MatLab, the following network structures are listed in table 1.

Table 1. Network Structures

No.	one-layered	No.	Two-layered.	No.	Three-layered.	No.	four-layered
1	1	7	1-1	13	2-2-2	19	3-3-3-3
2	2	8	2-2	14	3-2-2	20	4-3-3-3
3	3	9	3-2	15	4-3-2	21	5-4-3-3
4	5	10	5-2	16	5-3-2	22	8-5-4-3
5	8	11	8-4	17	8-3-2	23	10-8-4-3
6	10	12	10-4	18	8-4-2	24	10-8-5-4

As mentioned, the NN input parameters are defined by the force and torque. On the other side, the grasp position conditions (normal, collision, obstruction or front collision) have been taken as an output parameter. Artificial neural network has to be trained with corresponding data in order to learn the functional relationships between input and output data pairs.

In this paper, each one of these 24 different network structures has been trained with two training algorithms: Levenberg-Marquardt (LM) [13, 14] and Elman respectively. The sigmoid activation function has been used in hidden layers (1) and linear in output layers (2).

$$f(x) = \frac{1}{1+ e^{-x}}. \quad (1)$$

$$f(x) = x\,. \tag{2}$$

In addition, The NN performance is evaluated using the MSE (3).

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(\mathrm{T_i} - \mathrm{Y_i})\,. \tag{3}$$

Where N is number of data, T is the target and Y is the output. Figure 2. Presents an example of LMFF training implantation in Matlab® software

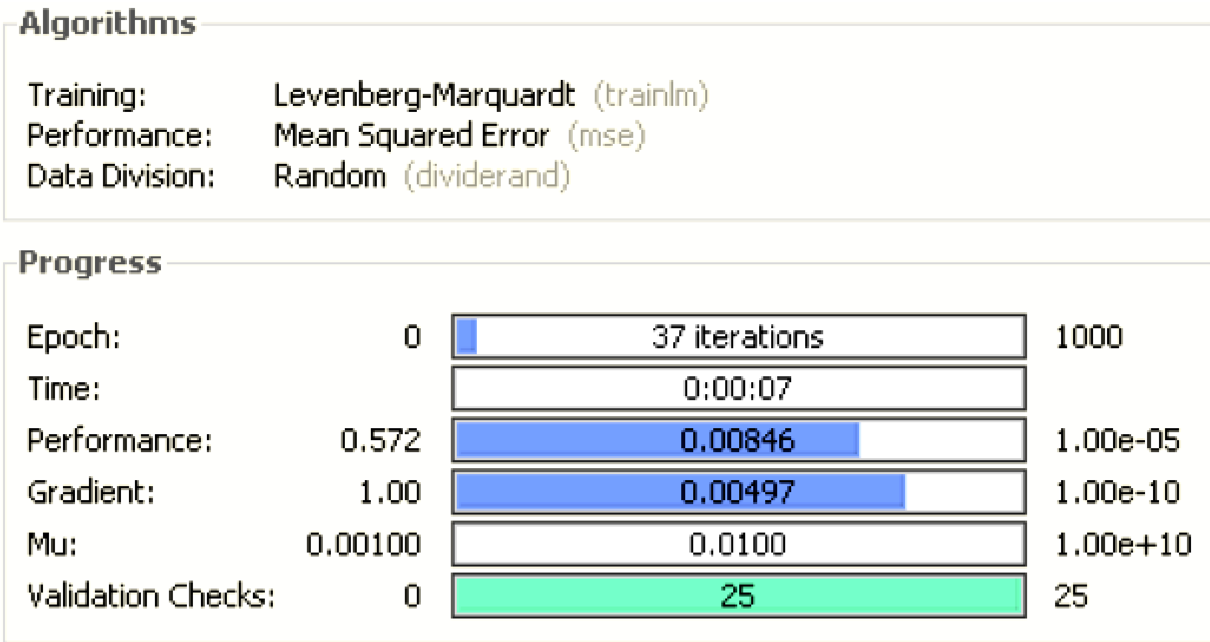

Figure 2. LMFF Training In MATLAB®

4. Results and Discussion

Based on the networks' capabilities, it is possible to find which network structure shows the performance results versus influence of the learning algorithm. The Matlab is used for implementation and testing. In order to find optimal NN, The total number of 24 different structures and architecture has been tested several times, In order to find optimal NN to evaluate their capabilities for generalizing the performance under different conditions.

The testing results in terms of MSE on test data which is 30% of the dataset for LMFF and Elman RNN in are given in Figure 3 and figure 4, respectively. The NN structurers in the figures

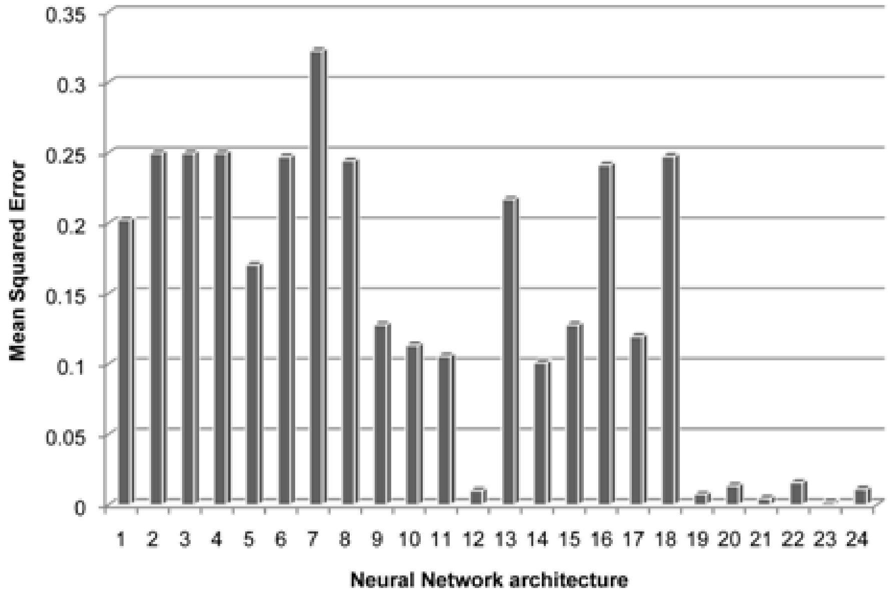

Figure 3. LM Feed forward *Testing*

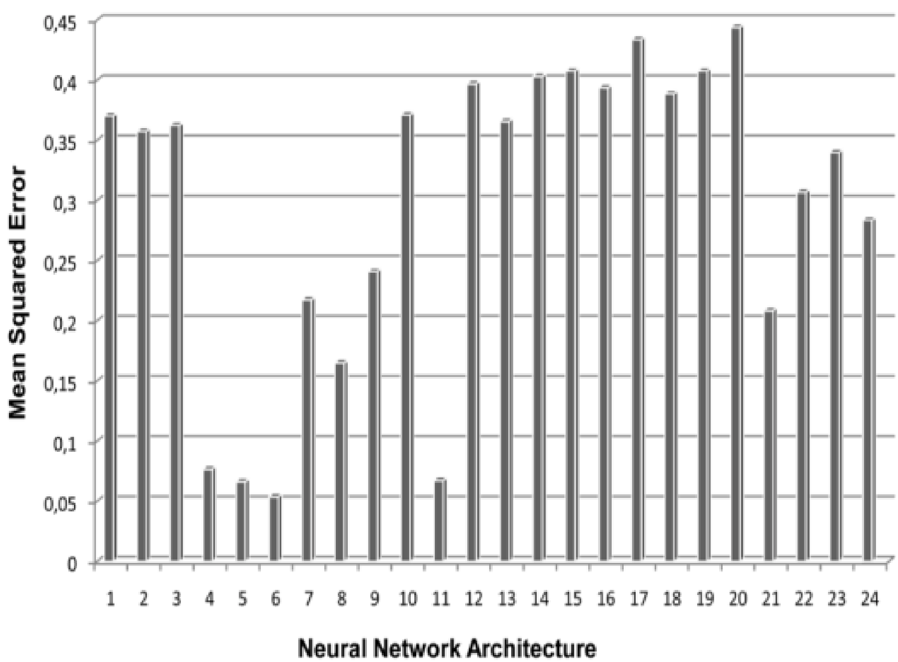

Figure 4. ELMAN *RNN Testing*

represent network number given in table 1.The testing results by means of Mean Squared Error (MSE) for (LMFF) and Elman (RNN). It is obvious that the MSE for LM has the decreasing trend when number of neurons and layers increases. In other words, the larger number of neurons and layers has positive influence on the training process . As it is obvious from Figure.3 and Figure.4, the (LMFF) showed overall better results. Smallest MSE was reported for structure No. 23 in Table 1, and the MSE is 0.0023.

In the case of Elman Neural Network, the similar conclusion can be obtained. Overall, the NN with 1 hidden layers and show the best performances. Particularly, in the case of Elman algorithm, the network number 6 (table 1) has the smallest test MSE and was reported for ([10]) structure (is equal 0.0539).

Figure 5 shows, respectively, screen captions of the LM NN training windows obtained using the toolbox in MATLAB®. The output tracks the targets very well for training, testing, and validation, and the R-value is over 0.96 for the total response. and for the Elman NN ,the validation and training plots are shown in Figure 6. The LMFF show overall better results than Elman RNN. Smallest value of MSE was reported for [10-8-4-3] structure (see Table 1).

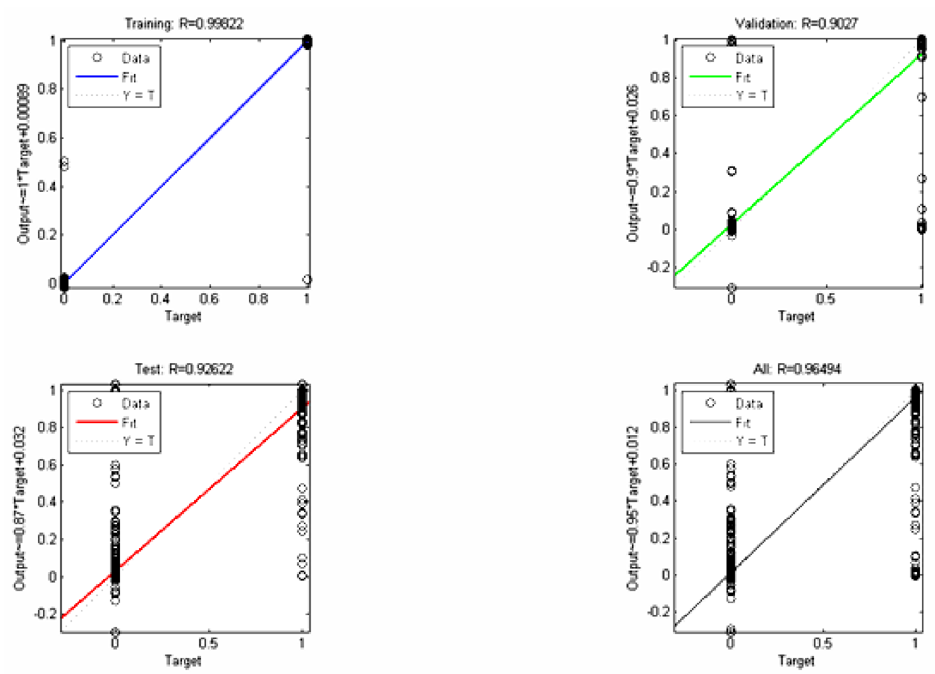

Figure 5. LM FF Training Windows

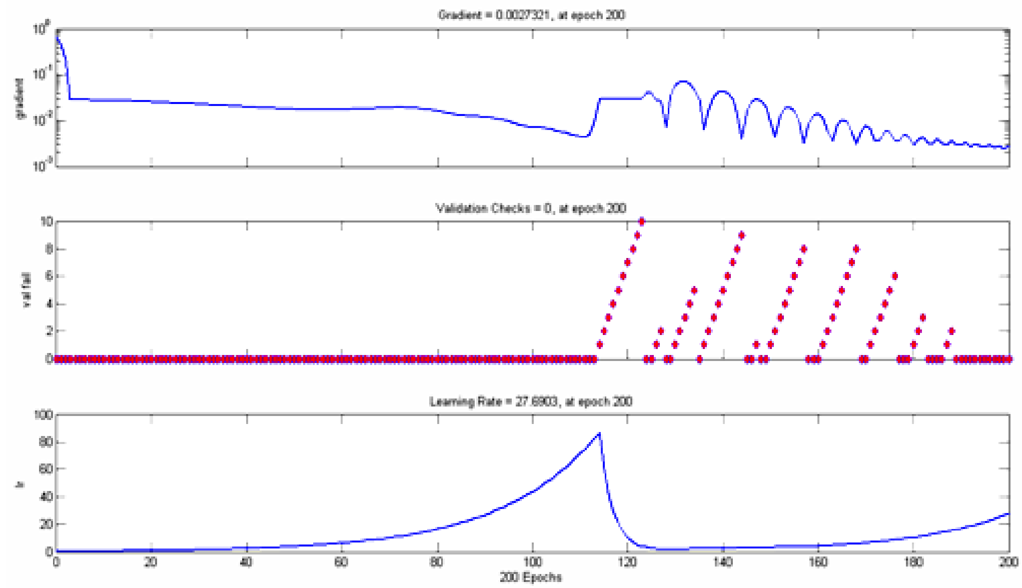

Figure 6. Validation And Training Of Elman RNN

5. Conclusions

This paper has shown the properties of two types of ANN, namely feed- forward NN and recurrent NN that have been trained with the real data to evaluate the performance and ability of ANN. In this papar , The efficiency of the Elman RNN was compared with the MLFF (multi-layer feed forward) .Asuccessful mapping from the execution forces and torques to the 4 possible cases that correspond to the particular input (normal, collision, obstruction or front collision) is developed using NN. The training dataset consists of a real system of robot that is recorded immediately during the execution of the specific task. In order to fully show the robustness of the approach. Results through networks show usefulness and the applicability for the evalution of the proposed approach in terms of training error.

References

[1] L .P. VeeLenturf. Analysis and applications of artificial nural networks .Prentice Hall international Uk Ltd. 1995

[2] G. K. Venayagamoorthy , "A Successful Interdisciplinary Course on Computational Intelligence", *IEEE Computational Intelligence Magazine* - *A special issue on Educational Issues on Computationa lIntelligence*, Feb. 2009, pp. 14-23.

[3] Krose, B. and van der Smagt, P .. An Introduction to Neural Networks, 8th Edition, The University of Amsterdam , Amsterdam , 1996

[4] M, T, Hagan, H,B. Demuth, M. H, Beat, Neutral Networks Design, PWS ,Beston, 2005

[5] Li, Y., Powers, D. and Wen, P.: An improved training approach for feedforward neural networks. http://david.wardpowers.info/Research/AI/papers/200002-sub-LSBNN.pdf

[6] Murat Kayri. "Predictive Abilities of Bayesian Regularization and Levenberg–Marquardt Algorithms in Artificial Neural Networks": A Comparative Empirical Study on Social Data. Math. Comput. Appl. 2016

[7] Salihu . ,A. , Garko**. "**Evaluating Feedforward And Elman Recurrent Neural Network Performances In Time Series Forecasting" Indian Journal of Pure and Applied Mathematics · June 2015

[8] Richard L. W, M. R, and Venayagamoorthy. "Comparison of Feedforward and Feedback Neural Network Architectures for Short Term Wind Speed Prediction". International Joint Conference on Neural Networks, Atlanta, Georgia, USA, June 14-19, 2009

[9] Elman, J, L. Finding structure in time. Cognitive Science,14,179 211.[Cited on p.48.],1990

[10] M. Jordan, "Serial order: A parallel distributed processing approach, "*Adv. Psychol.*, 1997

[11] D. Tank and J. Hopfield, "Neural computation by concentrating information in time,"*Proc. Natl. …*, 1987.

[12] http://www.ics.uci.edu/~mlearn/MLRepository.html.

[13] Fun, M.-H. and M. T. Hagan," Levenberg-marquardt training for modular networks,"*Proceedings of the 1996 International JointConference on Neural Networks,* 468{473, 1996

[14] Hagan, M. T. and Menhaj M. B., "Training feed forward networks with the Marquardt algorithm", *IEEE Transaction on Neural Networks*, **5**(6), 1994, pp 989-993.