Performance Evaluation of Optimised Backpropagation Algorithms for Yorùbá Character Feature Extraction and Recognition

Amos Orenyi Bajeh[1], Muftau Oluwatosin Wasiu[2], Fatima Enehezei Usman-Hamza[3]
[1, 2, 3] Department of Computer Science
University of Ilorin, Ilorin, Nigeria
Email: bajehamos@unilorin.edu.ng

***Abstract*— Character recognition has been an important area of research in the last few decades. It is basically divided into two major types namely online and offline (handwritten) character recognitions. Characters with tonal marks (diacritics) such as Yorùbá characters (orthography) had posed more challenges than their counterparts with no tonal marks and as a result require some optimization methods to improve the recognition rate and reduce the error rate. This study evaluated the performance of four optimized backpropagation algorithms, Levenberg-Marquardt, Quasi-Newton BFGS, Resilient Propagation and Scaled Conjugate Gradient, on Yorùbá character recognition. The method used in this study involves the five basic stages of image processing namely; image acquisition, image preprocessing, segmentation, feature extraction and classification. The performances of the algorithms were experimentally measured using mean squared error (MSE), epochs, accuracy and response time. From the experiments, it was observed that the Levenberg-Marquardt training algorithm has the best accuracy of 98.8%; Resilient Propagation and Scaled Conjugate Gradient are the fastest to converge with an average response time of 2 seconds. The results obtained can serve as a fundamental guideline in adopting the most relevant training algorithm for character image recognition.**

Keywords— *character recognition, optimized backpropagation algorithms, Yorùbá characters, Neural Network*

1. INTRODUCTION

Character recognition as an area in the field of pattern recognition has been an interesting and researchable field over the last couple of decades. In fact, character recognition has found its usefulness in several application areas such as reading of bank cheques, deciphering zip codes, document analysis and retrieval.

According to Mahmood et al. [1], Ibrahim and Odejobi [2] Character recognition can be divided into two: (1) online character recognition in which text is automatically converted as it is written on digitizer such as PC tablets, where a sensor picks up the pen velocity as characters are scripted. The signals obtained are transformed into a letter code which is usable to computer and text processing applications. (2) Offline character in which handwritten characters are scanned in form of paper document, processed and converted to binary or grayscale to make available to a recognition system. However, most of the research efforts have been focused on the recognition of English and Latin characters; little work has been done on African characters [3, 4, 5].

Back propagation is one of the most widely used learning algorithms to solve many classification problems by using the concept of multilayer perceptron (MLP) training and testing. However, it comes with its shortcomings which are slow convergence and getting trapped in the local minima. But many solutions have been proposed by many neural network researchers to overcome the problem of slow convergence and hill climb the local minima to reach the optimal solution (global minimum). In this view, many powerful numerical optimization algorithms have been devised, most of which have been based on gradient descent algorithms [6] such as conjugate gradient algorithms, Quasi-Newton and Resilient propagation (RP), Levenberg-Marquardt (LM) etc. Meanwhile, there is need to perform an experimental evaluation of some of the most promising optimization algorithms to decide which training algorithm is the best in practice for the research. Also, such evaluation needs to be performed in the context of other character types -other than English & Latin- that poses more challenges in processing.

The main goal of the study is to evaluate the performance of optimized back propagation algorithms for recognition of Yorùbá handwritten characters. Four training algorithms: Scaled Conjugate Gradient (SCG), Resilient Propagation (RP), Quasi-Newton BFGS and Levenberg-Marquardt (LM) were investigated. Their performance was evaluated based on Mean Square Error (MSE), Epochs (no of iterations), Speed (time) and Accuracy.

2. RELATED WORKS

A. Character Recognition Models

Ebenezer et al. [7] in their work proposed for recognition of Igbo vowel characters using Artificial Neural Network. A standard back propagation with adaptive learning rate and adaptive momentum were used and recognition rate of 90.2% was obtained after testing the neural network with 30% of their dataset. Nonetheless, their research was restricted to 9 vowel characters of the Igbo orthography.

Abdulrahman and Odetunji (2011) developed a technique for the classification of diacritically marked uppercase Yorùbá letters in offline mode. The system involves six stages. They built Bayesian stage using 40 samples per Bayesian class. The system was tested in two folds. In the first fold, they tested it on eight independent samples of each of the seventeen classes of the diacritical letters. A recognition rate of 91.18 % was obtained. In the second fold, the system was tested on three non-independent samples of each of the six Bayesian classes and a recognition rate of 94.4% was recorded. The work presented an approach of Bayesian rule and decision tree. However, their work did not cover the entire Yorùbá orthography which also includes lowercase letters.

Oyebade et al. [8] in their work, investigated the tolerance in neural network based recognition systems to some common pattern variances that occur in pattern recognition. Handwritten Yorùbá vowel characters were used to evaluate the performance of deep learning networks considered in their research. The research basically focused on comparing error rates or noise for the network architectures employed. It was observed that the pre-trained deep networks outperformed the untrained shallow networks of Back propagation Neural Network (BPNN). However, researches have shown that none of deep learning models work as classification algorithms per se. instead, they are used for pre-training – learning transformations from low-level and hard-to-consume representation (like pixels) to a high-level one. Once deep (or may be not deep) network is pre-trained, input vectors are transformed to a better representation and resulting vectors are finally passed to real classifiers such as SVM, ANN or Logistic regression.

On the contrary, Joarder and Aziz [9] in their paper presented a neural based invariant character recognition system using double back propagation algorithm. Their system was tested with English numeric digits. The test involved rotated, scaled and translated versions of exemplar patterns. The system successfully recognized 97% of the tested patterns. Nonetheless, the system was not used to test English alphabetic characters, either printed or handwritten and other similar applications.

Iorundu and Esiefarienrhe [10] developed an artificial Neural Network model for Tiv character recognition (ANNTCR). They employed Macromedia Fireworks 8 to resize, crop and clean the image of each character in order to reduce noise. The system was designed and implemented using Java Programming language on Econg Framework. Both the training and testing were done using feed forward resilient propagation neural network. Further, the system was tested with characters of different font styles. The result showed that the average recognition rate of the system was 99.4% while the system's rejection rate was below 1%.

Similarly, Mahmood et al. [1] in their work developed an Artificial Neural Network model for Yorùbá Character Recognition (ANNYCR). Macromedia Fireworks application was used to resize, crop and clean the image of each character in order to reduce noise. Edge-end pixel extraction algorithm was employed to extract numerical data for character analysis. The ANNYCR was implemented using Java Programming language on Econg framework together with a well-known Artificial Neural Network model for English characters (ANNECR). The ANNYCR was trained using feedforward Resilient Supervised Back Propagation algorithm, while supervised BP was used for ANNECR. The result showed that ANNYCR recognized all Yorùbá alphabets including characters with dot, ligature and tonal sign, while ANNECR could not recognize Yorùbá characters with dot, ligature and tonal signs. During the training, it was observed that BP algorithm was not converging when used to train Yorùbá characters whereas Feedforward Resilient propagation converged when used to train both Yorùbá and English characters. The result showed that BP cannot be applied for Yorùbá character recognition as it can be used for recognition of English characters.

Odetunji [11] use frequency profiling to characterize three Yoruba tones in speech. The study developed multi-layer perceptron (MLP) and recurrent neural network (RNN) and compared their performances. The two models performed well with RNN outperforming MLP.

Oyebode and Kamil [12] studied the performance of convoluted neural network and its variants using Yoruba vowel characters. The result of the study showed that convolutional neural network (with convolutional encoder) performed better that Backpropagation neural network, deep belief network and others.

B. Comparative Studies on Training Algorithm

Bhavna and Venugopalan [13] presented a study on comparative analysis of neural network training functions for hematoma classification of brain CT images using Gradient Descent BP, Gradient Descent Momentum (GDM), resilient BP, Conjugate Gradient BP algorithms and Quasi-Newton based algorithms. The research compared the training algorithms on the basis of Mean Squared Error (MSE), recognition accuracy, rate of convergence and correctness of the classification. In the study, no significant differences were found among the correct classification percentage for Resilient BP, Scaled Conjugate BP and Levenberg-Marquardt algorithms, all are in acceptable range. Meanwhile, the convergence speed of Levenberg-Marquardt and Scaled Conjugate algorithms are found to be higher than other training functions. Based on Epochs (iterations) and MSE parameters, Levenberg-Marquardt and Scaled Conjugate algorithms outperformed other training functions.

Bhavani et al. [14] described a method for the classification of respiratory states based on four significant features using Artificial Neural Network (ANN). They analysed the performance of five back propagation training algorithms namely; Levenberg-Marquardt, One-Step Secant, Powell-Beale Restarts, Quasi-Newton BFGS, and Scaled Conjugate Gradient for classification of respiratory states. In their experiment, it was observed that Levenberg-Marquardt algorithm was the best in approximately 99% of the test cases.

Haradhan et al. [15] in their own study performed some tests on some well-known training algorithms: Levenberg-Marquardt, Resilient Propagation and Scaled Conjugate Gradient to evaluate their performances for scene illumination classification. The result of their study shown that Levenberg-Marquardt is the most accurate with recognition rate of 94.41% and Resilient Propagation as the fastest method with response time of 0.426 seconds.

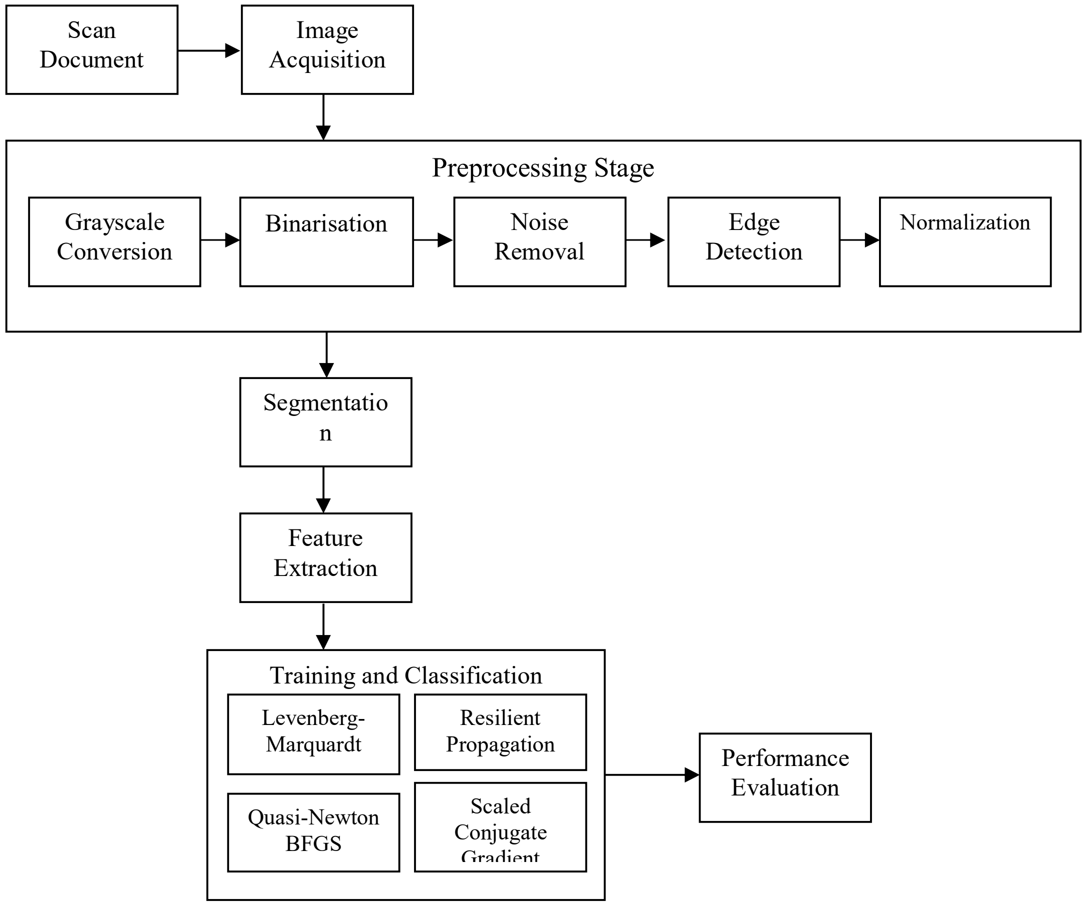


Fig. 1. Overview of methodology

3. METHODOLOGY

The overview of the methodology used in this study comprises of five stages as shown in Fig. 1.

A. Image Acquisition

This is the first stage in the process of character recognition. Handwritten characters were collected from volunteers and indigenous writers. The acquired data was scanned as input image to the preprocessing stage. The scanned data were saved as .jpeg file format and stored in a database. In this research, 20 samples were collected for each of the 15 characters considered.

B. Preprocessing

The output of the image acquisition phase serves as input to the preprocessing stage. The scanned image is loaded into MATLAB development environment for processing. To create a usable input vector containing features of a character, colour of an image is irrelevant. Thus, acquired image is first converted to grayscale, removing all colours and hue, and retained the luminosity. Binarisation which converted the grayscale image to binary forms of 0s and 1s using Otsu method was applied. Conversion to binary is necessary because, it helps in character segmentation.

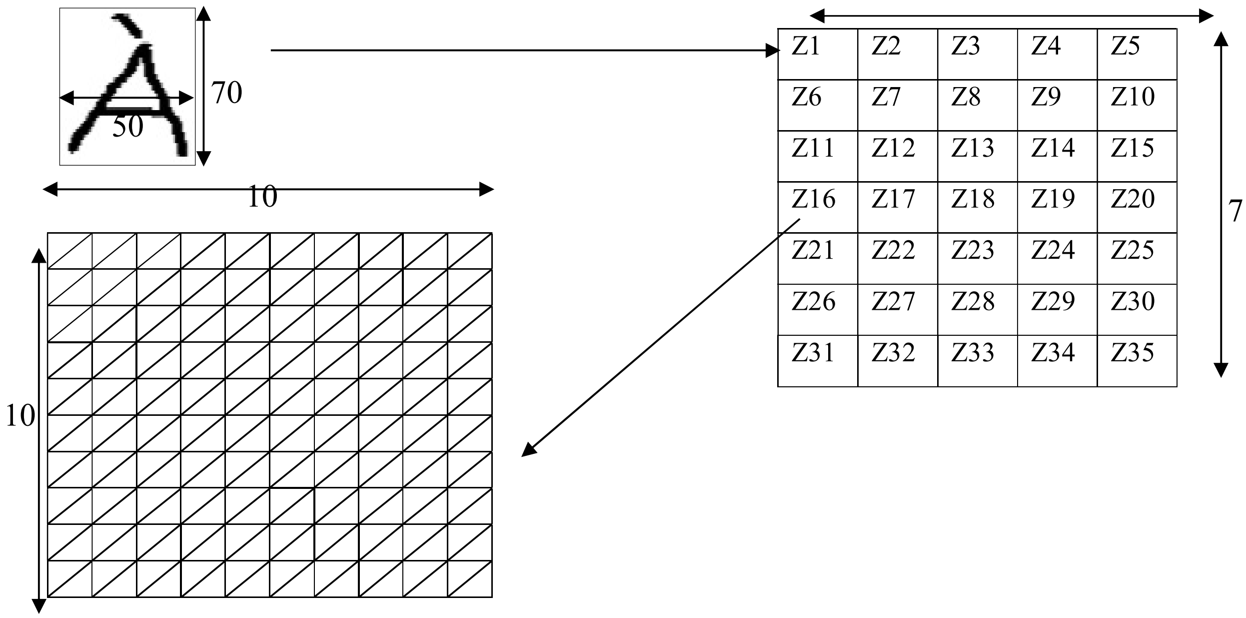
In the noise reduction process, noise caused by scanner, paper quality and image conversion is detected and removed, then connected components are reduced. In this study the adaptive filtering method, wiener filter, was implemented using MATLAB image processing toolbox as it helps to preserve edges, high frequency parts and does not blur the image like linear filter. Median filtering is also performed to further enhance the image quality by removing any leftover noise. Edges of the image were also detected to help in the process of labeling for segmentation.

C. Segmentation

In this study, the preprocessed input image is segmented into isolated characters by assigning a number to each character using a labeling process. This labeling process provides information about number of characters in the image. For each character, minimum and maximum row, along with columns are used to determine the starting pixel of the character. These values are assigned as a row vector. Each individual character is uniformly resized into 70 X 50 pixels (normalization).

D. Feature Extraction

In this study, Diagonal based approach was adopted to extract the features of the characters. The resized character of size 70x50 pixels is further divided into 35 equal zones, each of size 10x10 pixels. The features are extracted from the pixels of each zone by moving along their diagonals. This procedure is repeated for all the zones leading to extraction of 35 features for each character. These extracted features were used to train a multilayer feed-forward neural network. Each zone has 19 diagonal lines and the foreground pixels along each diagonal line are summed to get a single sub-feature. Thus, 19 sub-features are obtained from each zone. These 19 sub-feature values are averaged to form a single feature value and placed in the corresponding zone (Fig. 2). This procedure is sequentially repeated for all zones. There could be some zones whose diagonals are empty of foreground pixels. The feature values corresponding to these zones are zero. Moreover, 7 and 5 features are obtained by averaging the values placed in zones row-wise and column-wise respectively [17].



Fig, 2. Diagonal Feature Extraction

E. Training and Classification

In this study, Artificial Neural Network (ANN) was used for classification of characters. The optimized back propagation algorithms were used in training the Artificial Neural Network in order to minimize the limitations identified with conventional back propagation. Each training algorithm was tested on average of five trials on the dataset and results were evaluated by recording the classification accuracy, MSE, epochs and response time. During the experiments, the network parameters were kept constant for each training algorithm in order to ensure equal opportunity for evaluating the algorithms. For the classifier in this study, a feed forward ANN was experimentally designed to perform pattern recognition which classifies input vectors into 15 classes of tonal characters of Yorùbá orthography. The architecture of a typical 2-layer ANN is shown in Fig. 3.

As illustrated in Fig. 3, p is the input vector of R x 1 dimension where R is the number of rows, W is the weight matrix of dimension S x R where S is the number of neurons in the layer, b is the bias vector which is a weight with 1 as input, n is the weighted input into the transfer function, a is the hidden layer output vector and y is the output vector from the network. The designed network in this study has 35 input neurons (corresponding to 7 x 5 matrices) and 15 output neurons because there are 15 target values which represent the 15 classes associated with each input vector. When an input vector is entered into the network, the corresponding neuron produces a 1 and the others produce 0s.

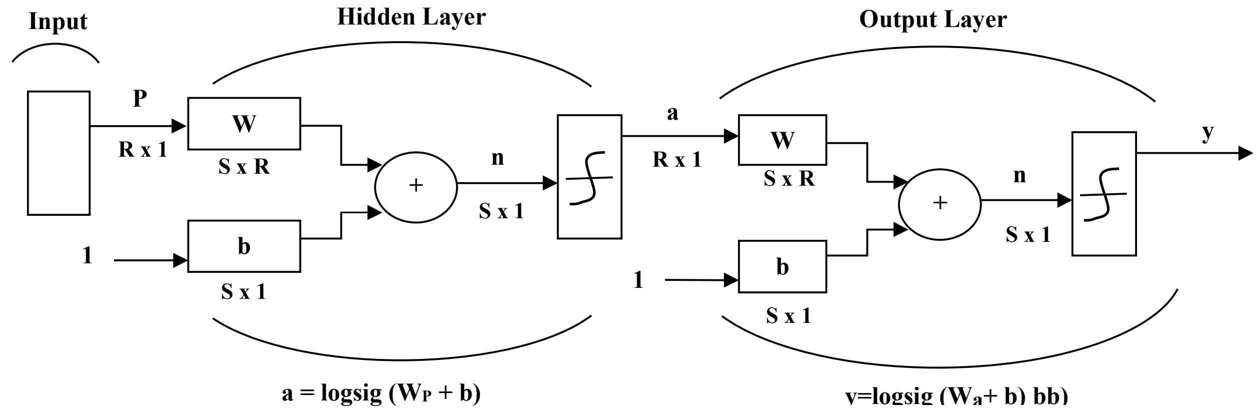


Fig. 3. Architecture of a 2-layer Feedforward ANN

F. Performance Evaluation

The performances of the training algorithms in the classification process were evaluated by the following indices;

MSE: Mean square error is computed using the following equation:

$$MSE = \sum_{i=1}^{n} \frac{(a_i - o_i)^2}{n} \quad \ldots \quad (1)$$

where a_i is the target value, o_i is the observed network output and n is the total number of dataset.

Accuracy: this is the ratio of the number of correctly classified samples by the total number of samples. It is computed from a confusion matrix.

Response time is the time at which the network model converges to its optimum.

Epoch is a measure of the number of times all of the training vectors are used once to update the weights.

G. Experiments

This study comprises of five experiments, Experiment 1 – 5, conducted to measure the performances of the four training algorithms:

Experiments 1 was aimed at making a choice of the optimized back propagation (BP) training algorithm to be further studied for the classification process. In this experiment the four training algorithms were evaluated with varying number of hidden layer of neurons from 10 to 50 in steps of 10. Five different trials were run for each training algorithm and the epoch and mean square error (performance measure) for each trial were recorded.

The aim of Experiment 2 is to obtain the optimal number of epochs for training the ANN. That is, this experiment determines the epochs of the selected algorithm from Experiment 1 for training the ANN designed in this study. Five different trials with number of hidden layer neurons varied from 10 to 50 in steps of 10 and epochs varied from 50 to 200 in steps of 50 were applied on the best algorithm returned by Experiment 1.

The goal of Experiment 3 is to determine the rate of convergence (no of epochs) at which the performance goal of 0.001 MSE will be achieved by the algorithm selected from Experiment 1. Using 50 hidden layer of neurons and the selected algorithm, the epoch was varied to determine the optimal performing epoch for the algorithm on the ANN.

The motive of the fourth experiment, Experiment 4, is to obtain the appropriate transfer (activation) functions for the hidden and output layers of the ANN using the training algorithm selected in Experiment 1 and the other parameters determined in the other experiments (i.e., Experiments 2 and 3). The performance of the ANN using different transfer function combinations, tansig/logsig, logsig/logsig, tansig/tansig and logsig/tansig activation functions, in the hidden and output layers was measured.

Experiment 5 uses the selected algorithm in Experiment 1 and all the parameter values determined in Experiments 2, 3 and 4 to develop the proposed ANN for Yorùbá character recognition. This ANN is expected to be the optimal model for the recognition of Yorùbá characters

4. RESULTS AND DISCUSSION

In this study, four experiments were carried out to evaluate the performance of the selected training algorithms and adopt the best for the optimal result for classification.

Experiment 1

The results obtained from this experiment for each of the training algorithms is shown in Fig. 4. Fig. 4 illustrates that Levenberg-Marquardt training algorithms produced the best performance across varying number of hidden layer neurons for the number of trials. Hence, Levenberg-Marquardt training algorithm is adopted for the other experiments in this study. That is, in order to ascertain the efficacy of the Levenberg-Marquardt algorithm, the other three (3) experiments were carried out on it.

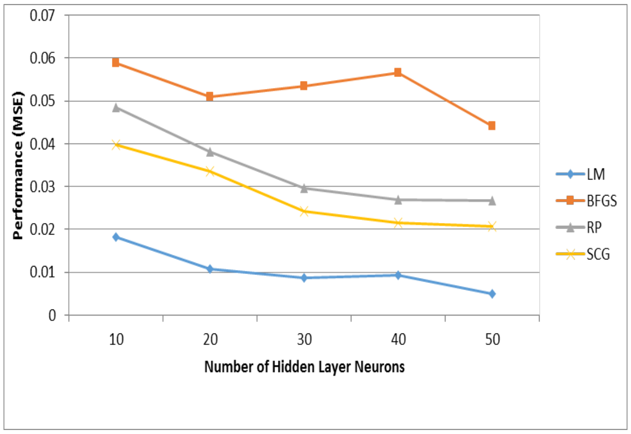


Fig. 4. Performance of the training algorithms at varied number of hidden layer

Experiment 2

The MSE of each of the varied epoch values of Levenberg-Marquardt algorithm (returned by Experiment 1) are presented in Fig. 5.

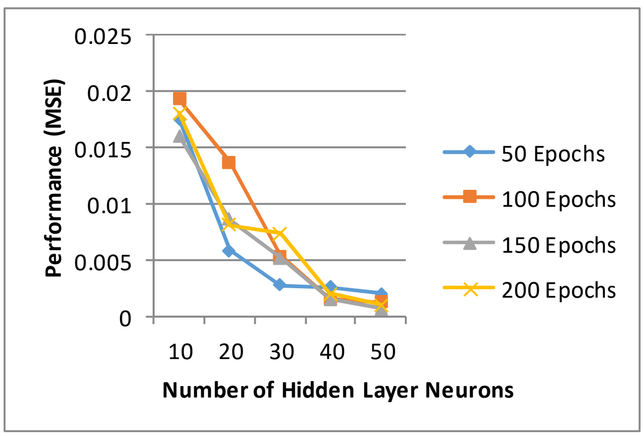


Fig. 5. Levenberg-Marquardt performance graph (MSE vs. number of hidden layer neurons for four different epochs (50, 100, 150, and 200))

Fig. 5 shows that 150 and 200 epochs achieved nearly similar level of performances for Levenberg-Marquardt algorithm while lower epochs have comparatively low performance measures. Hence, in order to maximize computational cost and response time and since both 150 and 200 epochs have nearly similar performances at 50 hidden layer neurons, 150 epochs was selected as one of the parameters for Levenberg-Marquardt algorithm in training the ANN.

Experiment 3

Using Levenberg-Marquardt algorithm (the best performing algorithm returned in Experiment 1) with 150 and 200 epochs, 50 hidden layer neurons; the performance of the ANN was tested for achieving the performance goal. The result of the trials in this experiment is shown in Fig. 6.

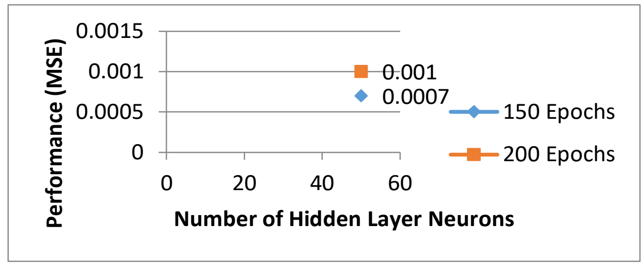


Fig. 6. Levenberg-Marquardt rate of convergence with 150 and 200 epochs

Fig. 6 shows that the performance (MSE) goal was met at minimal rate of convergence using 150 epochs in comparison with rate of convergence at 200 epochs. It is also noted that as the number of epochs increase, it affected the performance (MSE) of the ANN. Thus, 150 epochs is adopted for the training of the ANN for Yorùbá character recognition.

Experiment 4

Using Levenberg-Marquardt algorithm, epoch of 150, number of hidden layer neurons from 10 to 50 in steps of 10; the performance of the ANN using different transfer function combinations (i.e. tansig/logsig, logsig/logsig,tansig/tansig and logsig/tansig activation functions) in the hidden and output layers respectively. The performance result for different trials in this experiment is shown in Fig. 7.

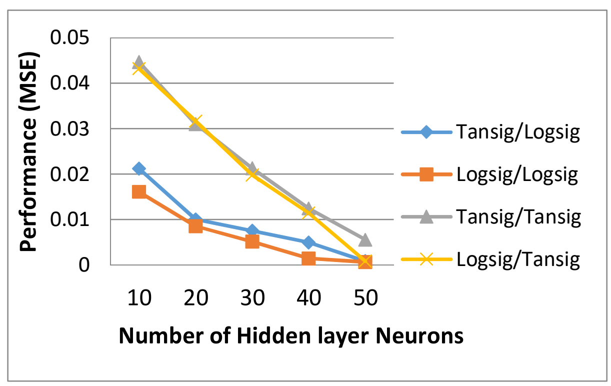


Fig. 7. LM performance plot against number of hidden layer neurons for different hidden and output layers activation functions

Fig. 7 shows a Mean Square Error (MSE) performance plot against number of hidden layer neurons for different transfer functions (tansig/logsig, logsig/logsig, tansig/tansig and logsig/tansig) in the hidden and output layers. The figure indicates that logsig/logsig and tansig/logsig MSEs are far less than that of tansig/tansig, and logsig/logsig performs better than tansig/logsig. Hence, logsig/logsig transfer functions were adopted for the hidden and output layers respectively for the ANN model in this study.

Experiments 1-4 have determined the best network parameter settings that can be used to achieve an optimal performance of ANN in the recognition of Yorùbá characters. Thus, Experiment 5 used the configuration presented in Table 1 for the ANN for Yorùbá character recognition process. The architecture and training interface of the ANN in MATLAB is also presented in Fig. 8.

TABLE 1. CONFIGURATIONS OF THE ANN

S/No	Parameters	Values
1	Backpropagation Training Algorithm	Levenberg-Marquardt
2	Maximum number of epochs	150
3	Hidden layer transfer function	Logsig
4	Output layer transfer function	Logsig
5	Number of hidden layer's neurons	50

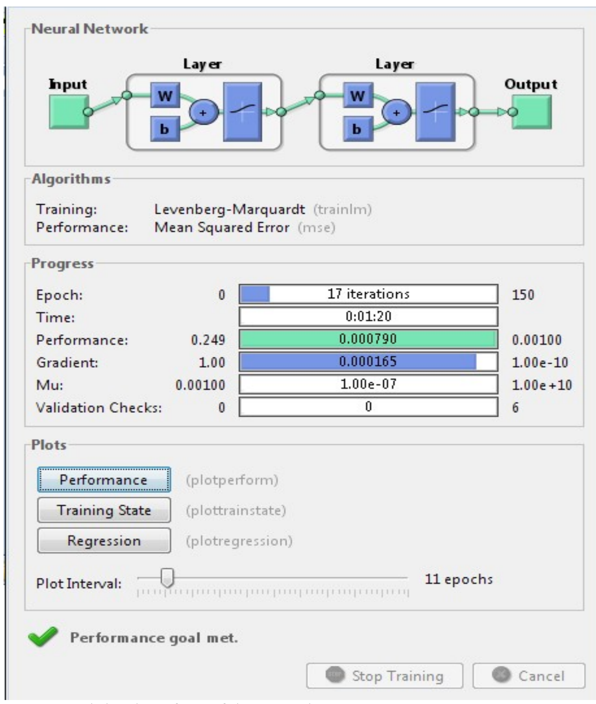


Fig. 8. Training interface of the ANN in MATLAB

The performance of the model based on the foregoing experimental results was evaluated with the MSE performance measure, classification accuracy using confusion matrix, rate of convergence (epochs), and response time. Fig. 9 presents the confusion matrix of the ANN. Confusion matrix also known as misclassification matrix is a table whose columns represent the target class and whose rows represent the output class. It is one of the useful tools used for analyzing a pattern recognition model. The correctly classified inputs show in the diagonal cells of the confusion matrix and the off-diagonal cells depict misclassification inputs. Table 2 show the optimal performance measures of the ANN.

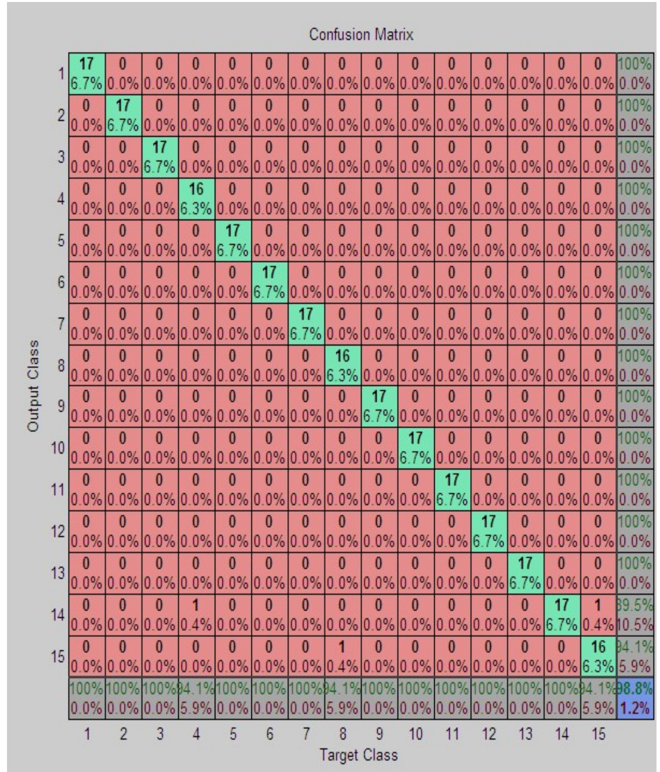


Fig. 9. Confusion Matrix showing Classification Accuracy

TABLE 2. PERFORMANCE MEASURES OF THE ANN

MSE	Accuracy	Rate of Convergence (Epochs)	Performance Goal
0.0007	98.8%	17 epochs	0.001

Furthermore, a regression analysis is done to see how close the ANN output is to the expected result. A regression analysis indicates the relationship between the outputs and the corresponding targets for the results of the model. Fig. 10 shows the regression analysis result. Three parameters are usually generated from regression analysis. The first two are the slope and the y-intercept of the best linear regression relating target to network outputs. The slope is 1 and the y-intercept is 0 if there is a perfect fit (i.e. output is approximately equal to target). The third variable is the correlation coefficient (R-value) between the output and the target, which is often used as a comparison index. If R-value is 1, there is a perfect correlation and perfect fit between the targets and the outputs. The result of this analysis shows that R-value is 0.992 which is closer to 1 and it means near perfect correlation between the ANN model's output and the expected result.

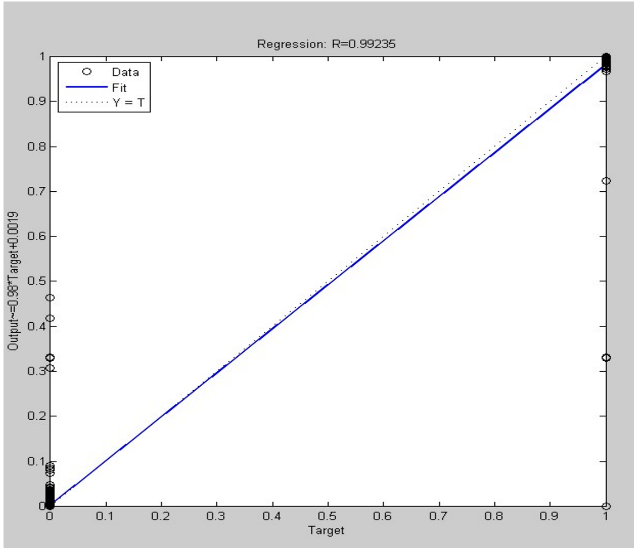


Fig. 10. Regression Analysis of the model

5. CONCLUSION

In this study, feed forward Artificial Neural Network was used for classification and can be adopted as the effective method for classification purpose with appropriate combination of training algorithms, transfer functions and other network parameters. The ANN was trained with Levenberg-Marquardt after it was selected as the better algorithm through several experiments using extracted feature sets from Yorùbá orthography. During the different experiments, it was found that rate of convergence of Levenberg-Marquardt was the best as well as lower MSE values which was approximately equal to a pre-determined performance goal of 0.001. A classification accuracy of 98.8 percent was recorded for the model with combination of logsig transfer functions at both hidden layer and output layer using 50 and 15 neurons respectively.

However, Levenberg-Marquardt can be combined with other training algorithms that are better in term of response time. This can help mitigate against intensive memory requirement of Levenberg-Marquardt. It should be noted that there was no improvement in the performance when the number of hidden layers was increased. In this regard, Neural Network ensemble and deep networks would be recommended for future study. The performance of the model improved with large datasets compared with few datasets. In this study, only uppercase characters with diacritics were considered, the model can be studied and enhanced with lowercase letters in future studies. Other variants of neural network will also be investigated further to compare performances within networks and also with other learning models [11, 12, 17].

REFERENCES

[1] Z.A. Mahmood, A.P. Adewole and A.T. Akinwale, "Artificial Neural Network Model for Yoruba Character Recognition." International Journal of the Nigeria Computer Society, 19(2), 32-37, 2012.

[2] A. O. Ibrahim, and O. A. Odejobi, "Towards Reliable Handwritten Character Recognition amid diacritic chaos", 2012, Retrieved from https://www.semanticscholar.org/paper/Towards-Reliable-Handwritten-Character-Recognition-Ibraheem-Odejobi/238c2f223f8ee03a1c049aa03887147c6b2d44c9

[3] S. Saurabh and P.S. Manu, "Performance Evaluation of Feedforward Neural Network with Soft Computing Techniques for Handwritten English Alphabets." International Journal of Applied Soft Computing, 11(1), pp. 1156-1182, 2011.

[4] V. Surabhi, C. Rashmi and T. Yogesh, "Character Recognition by Levenberg-Marquardt (LM) Algorithm Using Back Propagation ANN." International Journal on Recent and Innovation Trends in Computing and Communication, 2(8), pp. 2368-2372, 2014.

[5] M. Meshesha, "Recognition and Retrieval from Document Image Collections". Doctoral Thesis, International Institute of Information Technology, Hyderabad, India. 2008.

[6] M. Shashank and G.S. Tripatti, "Comparison of various Neural Networks Algorithms used for Location Estimation in Wireless Communication." International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2(6), pp. 2735-2741, 2013.

[7] O.O. Ebenezer, K.O. Oyebade, and A. Khashman, "Neural Network RecognitionSystem for Igbo Vowels." International Journal of Scientific and Engineering Research, 5(12), pp.435-440, 2014.

[8] K.O., Oyebade and O.O. Ebenezer, "Deep Learning in Character Recognition considering Pattern Invariance Constraints." International Journal of Intelligent Systems and Applications, Vol. 7 (7), pp. 1-10, 2015.

[9] K. Joarder and S.M. Aziz, "A Neural Network Based Character Recognition System Using Double Back Propagation." Malaysian Journal of Computer Science, 11(1), pp. 58-64, 1998.

[10] G.S. Iorundu, M.B. Esiefarinrhe, "An Artificial Neural Network Model for Tiv Character Recognition." Journal of Emerging Trends in Computing and Information Sciences, 6(10), pp. 573-577, 2015.

[11] O. A. Odelobi, "Recognition of Tones in Yoruba Speeach: Experiment with Artificial Neural Network" Studies in Computational Intelligence, 83. pp. 23-47. Springer. 2008.

[12] K.O. Oyebade and D. Kamil. "Patern Recognition: Invariance learning in convolitional auto encoder network". Int'l Journal Image, Graphics and Signal Processing, No. 3, pp. 19-27. 2016

[13] S. Bhavna, and K. Venugopolan, "Comparison of Neural Network TrainingFunctions for Hematoma Classification in Brain CT Images." IOSR Journal of Computer Engineering, 16(1), 2014, 31-35.

[14] A. Bhavani, K. Sankar Seethalakshmi, and D. Kumar, "Performance Evaluation of various Back-propagation Training Algorithms in the Neural Network based Classification of Respiratory Signals." International Journal of Neural Networks and Applications, 2011, 4(1), 31-36.

[15] Haradhan Chel, Aurpan Majumder and Debashis Nandi, "Scaled Conjugate Gradient Algorithm in Neural Network Based Approach for Handwritten Text Recognition", Communications and Computer and Information Science 204, pp. 196-210, 2011.

[16] J. Pradeep, E. Srinivasan and S. Himavathi, "Diagonal Based Feature Extraction For Handwritten Alphabets Recognition system Using Neural Network." International Journal of Computer Science & Informational Technology (IJCSIT), 3(1), pp. 27-38, 2011.

[17] M. Sheikhan. "Hybride of evolutionary and swarm intelligence algorithms for prosody modeling in natural spech synthesis". International Journal of Information and Communication Technology Research (IJICTR), Vol. 8 No. 2, pp. 33-44. 2016

