

Implementation of Optimal Hidden Neurons using a fuzzy Controller

Dr. Gauri Ghule^a, Dr. Prachi Mukherji^b, Dr. Pallavi Deshpande^c and Dr. Archana Ratnaparakh^d

^{a,c,d} E & TC Department, Vishwakarma Institute of Information Technology, Savitribai Phule Pune University, Pune, India

^b E & TC Department, Cummins College of Engineering for Women, Savitribai Phule Pune University, Pune, India

Article History: Received: 11 January 2021; Accepted: 27 February 2021; Published online: 5 April 2021

Abstract: Number of hidden neurons is necessary constant for tuning the neural network to achieve superior performance. These parameters are set manually through experimentation. The performance of the network is evaluated repeatedly to choose the best input parameters. Random selection of hidden neurons may cause underfitting or overfitting of the network. We propose a novel fuzzy controller for finding the optimal value of hidden neurons automatically. The hybrid classifier helps to design competent neural network architecture, eliminating manual intervention for setting the input parameters. The effectiveness of tuning the number of hidden neurons automatically on the convergence of a back-propagation neural network, is verified on speech data. The experimental outcomes demonstrate that the proposed Neuro-Fuzzy classifier can be viably utilized for speech recognition with maximum classification accuracy.

Keywords: backpropagation neural network, hidden neurons, neural network training, Neuro-Fuzzy classifier

1. Introduction

Backpropagation Neural Network classifier (BPNN) permits more complex, non-linear relationships of input data to output results [1]. The structure of BPNN classifier is affected by factors like the size of the training set, Input Layer (IL) size, Hidden layer (HL) size, activation function used for learning, Output Layer (OL) size, and so on. Out of these factors, we have dealt with Hidden Neurons (HN) in this research work. Some empirically derived thumb rules are in existence to set the hidden neurons. Each hidden neurons added increases the number of weights and impedes generalisation contributing to increased training time. The number of hidden neurons is gradually increased with repetitive training of the network and the Mean Squared Error (MSE) is monitored. This work proposes a technique to estimate the optimal value of hidden neurons using a fuzzy controller based on MSE. The fuzzy controller reduces the overhead of training the network repeatedly to find the optimal value of hidden neurons. It also avoids the necessity of checking the testing accuracy iteratively [2, 31]. The proposed Neuro-Fuzzy classifier is tested for accurate speech recognition. This paper is arranged as: Section 2 discusses related work, section 3 describes the proposed methodology, section 4 discusses results, and finally, the conclusion is given in Section 5.

RELATED WORK

Many applications of neural networks test a series of systems with an alternate estimation of hidden neurons, and the effective MSE is calculated. In [3], Li proposed a monotone index-based method to estimate the number of Hidden neurons in the three layer feedforward network. The method requires large dataset. It does not support multiple input or output.

Yuan et al. [4] have decided the hidden neurons size based on information entropy and decision tree algorithm. Initially, the neural network with random hidden neurons is trained by a set of training samples. Then the activation values of hidden neurons and information gain are calculated. A decision tree is used to overcome problems associated with higher number of neurons and problems such as shortage of capacity because of the few numbers of neurons are avoided. Boger et al. [5] suggested initialising the number of hidden neurons to number of input neurons. In case it doesn't produce sufficient accuracy, a number of output neurons are added to it. Researchers have proposed a couple of more thumb rules to choose the size of HL as, the number of hidden neurons ought to be not as much as twice of the IL neurons, the size of HL should lie between that of IL and OL [6,7].

Estimation of signal to noise ratio is used to optimize the number of hidden neurons while avoiding overfitting in the function approximation [8]. The criterion uses a small number of computations and can be used for other parameters of the network like number of iterations depending upon training error itself. The technique works well for small sized datasets. Review of methods to fix the number of hidden neurons is discussed in [9]. Authors have also proposed a new technique to fix the number of hidden neurons. They have examined 101 various hidden neurons fixation criteria to estimate the error in the neural network. The criteria with a minimum estimated error is fixed to train and test the network. The selected criterion is $(4n_2 + 3)/(n_2 - 8)$ where n is the number of input parameters. Panchal et al. [10] have reviewed various approaches to find the optimal value of hidden neurons. They include trial and error method with repeated attempts, rule of thumb method where predefined rules are used to decide the size of the HL, a simple

method where the number of hidden neurons is matched to the number of input and output nodes. Authors have experimented back propagation method and conjugate gradient method by increasing the number of HLs and the number of hidden neurons simultaneously. The combination that yields minimum MSE is used for experimentation.

METHODOLOGY

BPNN is a type of gradient descent algorithm [25]. Conventional BPNN implementation requires value of hidden neurons to be fed as input. If the hidden neuron value entered is too large, the network enters a saturation state [29,30]. The proposed classifier overcomes these drawbacks by autotuning the hidden neurons value.

The fuzzy controller determines whether the value of hidden neurons can be increased or decreased. In terms of linguistics the problem is considered. It can be mentioned that “if the MSE is too high at the output of one iteration, then the number of hidden neurons is reduced.” Conversely, “if the MSE is too low at the output of one iteration, then the number of hidden neurons increases.” Such laws suggest that there is a dead zone between the value of high and low hidden neurons that is “approximately right” for a network. The question is, what is ‘high’ and ‘low’ for the network and what is ‘about right’ for it. Fuzzy logic answers the questions using membership functions.

Figure 1 represents Neuro-Fuzzy classifier with hidden neurons tuning by the fuzzy controller. It provides a better framework for designing the neural network. The benefit of the approach is no manual setting of hidden neurons. It shows a traditional BPNN network with constant LR completes one iteration of the backpropagation algorithm and calculates MSE. The fuzzy controller provides the change required in the HN value ‘ΔHN’. It modifies the HN by repeating the process until ‘ΔHN’ reaches zero. The number of HN, when ΔHN reaches zero, is the desired near-optimal value. BPNN corrects MSE by back-propagating it while adjusting weights through the HL and back to the IL. The membership function of the fuzzy controller built for tuning the value of hidden neurons is represented in Fig.2. Y axis has degree of membership plotted and the other axis represents MSE. ‘Low and high’ are the linguistic labels used. Two segments are defined as shown in Fig.2. The two membership functions “low(MSE)” and “high(MSE)” are determined for a particular value of MSE. If the membership value is too high, value of hidden neurons is reduced. If the membership value is too low, the value of hidden neurons is increased.

The fuzzy controller defines fuzzy rules as,

$$\begin{aligned}
 \text{low(MSE)} &= \\
 \{ & \\
 1, & \text{ If } \text{MSE} \leq 0.01 \\
 (0.45 - \text{MSE}) / 0.44, & \text{ If } \text{MSE} > 0.01 \ \&\& \ \text{MSE} < 0.45 \\
 0 & \text{ If } \text{MSE} \geq 0.45 \\
 \} & \\
 \text{high(MSE)} &= \\
 \{ & \\
 0, & \text{ If } \text{MSE} \leq 0.45 \\
 (\text{MSE} - 0.45) / 0.020, & \text{ If } \text{MSE} > 0.45 \ \&\& \ \text{MSE} < 0.475 \\
 1 & \text{ If } \text{MSE} \geq 0.475 \\
 \} &
 \end{aligned}$$

The number of hidden neurons is tuned using following steps.

- In a BPNN with correct response as ‘ t ’, MSE is computed using (1) at the end of feed forward pass. ‘ u_i ’ represents the activation value of each cell.

$$\text{MSE} = \frac{1}{2} \sum_i^k (C_i - u_i)^2 \tag{1}$$

- For the first iteration, the number of output neurons is considered to be number of hidden neurons. The fuzzy membership function gives corresponding degree of membership for computed MSE. The required change in number of hidden neurons, ΔHN is computed using (2). As shown in (3), the value ΔHN is added to the number of hidden neurons in current iteration to get the corresponding value for next iteration. L is a tunable scalar parameter which is considered equal to 10 empirically.

$$\Delta HN = ((\text{low}(\text{MSE})D_{HN}) - (\text{high}(\text{MSE})D_{HN})) \tag{2}$$

$$\text{HN}(\text{next}_{itr}) = \text{HN}(\text{cur}_{itr}) + \Delta HN \tag{3}$$

- Above two steps are repeated till the ΔHN becomes zero. This final value of hidden neurons is considered as optimal number of hidden neurons.

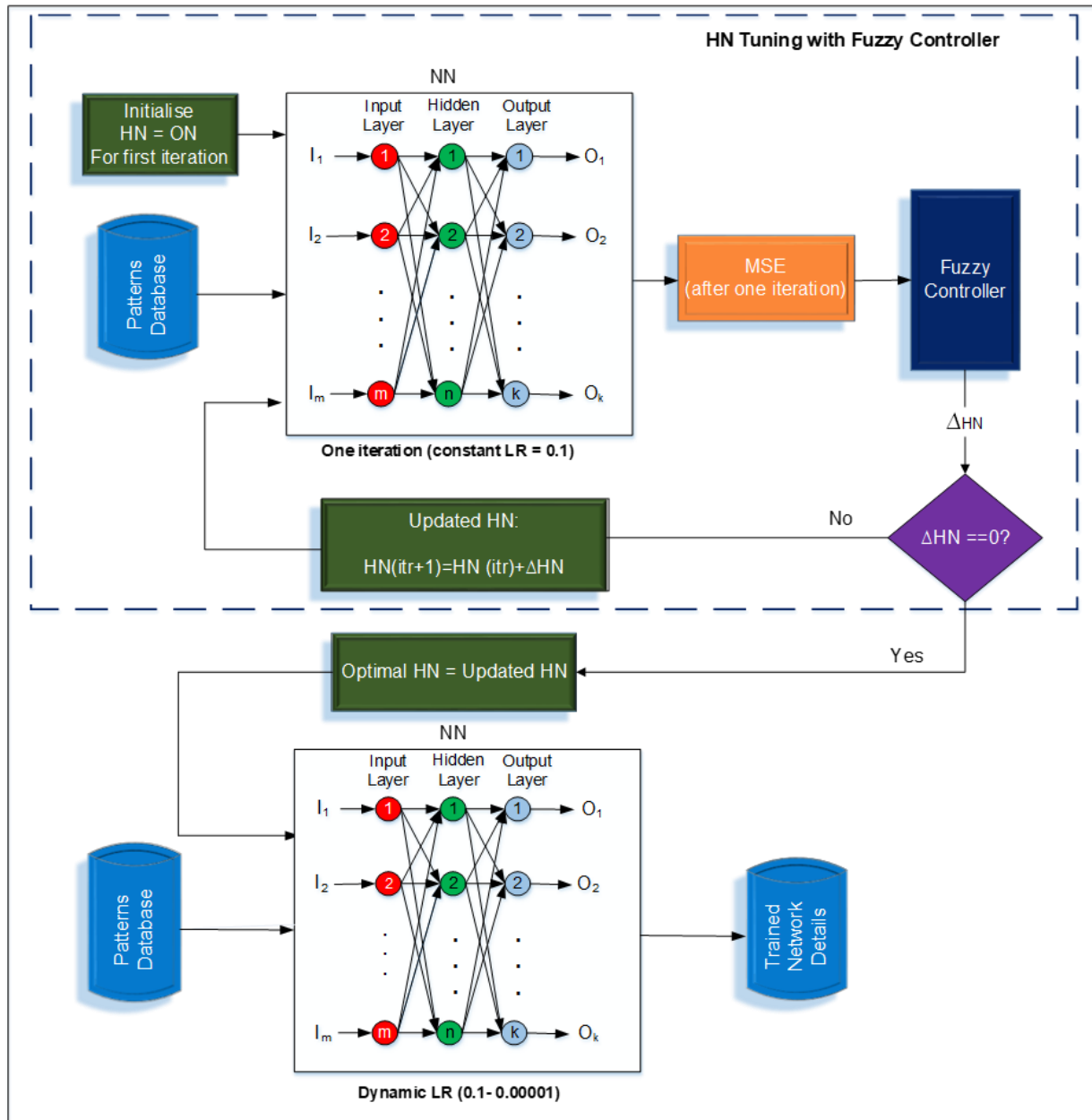


Figure 1 Neuro-Fuzzy classifier

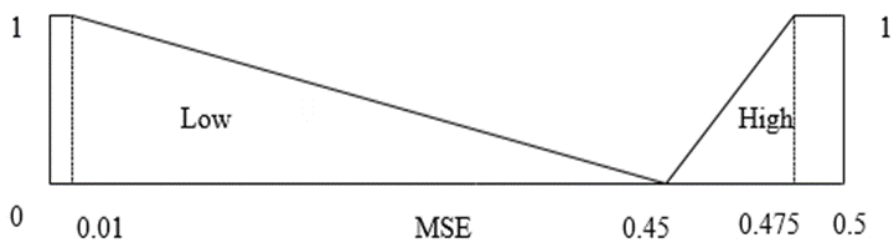


Figure 2. Membership function

The experimentation done for speech dataset provides highest results compared to state of the art approaches.

RESULTS

Ability to generalise and provide proper predictions for unknown situations is the strength of NN [26]. Backward error propagation is the most popular learning algorithm in artificial neural networks, where an error in the output is corrected by back-propagating it [24]. Existing hidden neurons finding methods determine the number of hidden neurons using several trials. The method starts with undersized hidden neurons and then add neurons to it. This approach is time-consuming and does not assure fixing the number of hidden neurons to its optimal value. The results of experimentation for speech recognition with the proposed hybrid Neuro-Fuzzy classifier approach are discussed in this section. The following section explains the speech datasets used for training and testing the network.

Three different speech datasets are used to analyze the performance of the proposed technique. The first dataset is Spoken Marathi Numerals Dataset (SMND) [11]. SMND is our own recorded dataset. The other datasets are TIDIGITS dataset [12] and Vowels Dataset (VD) [13]. Separate training and testing TIDIGITS datasets are available from the course ‘Fundamentals of speech recognition’. Different parameters for the three datasets are described in Table 1.

Table 1. Speech Datasets

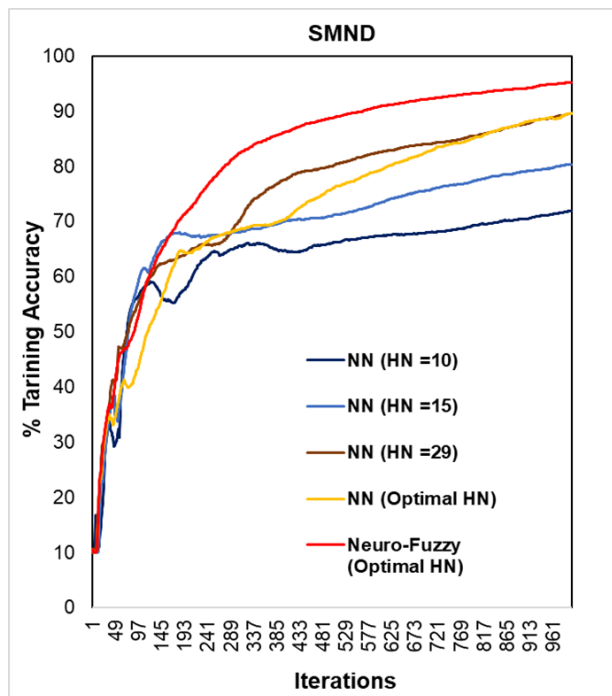
Parameter	SMND	TIDIGITS	VD
Author	Kondhalkar et al. [11]	Texas Instruments [12]	Hillenbrand et al. [13]
Number of Samples	10000	4554	1668
Sampling Frequency	16 kHz	8 kHz	16 kHz
Number of speakers	100	207	139
Vocabulary size	Marathi language Numerals “Shunya” to “Nau”	English language Numerals 0 to 9, ZERO	English Language Vowels
Output Classes	10	11	12
Utterances per class	10	2	1
Training Samples	6250	2068	900
Testing Samples	3750	2486	768

For the proposed Neuro-Fuzzy approach, the number of hidden neurons is tuned automatically. The training is stopped for minimum error or a defined number of iterations. The parameters designed for BPNN used for three datasets are listed in Table 2.

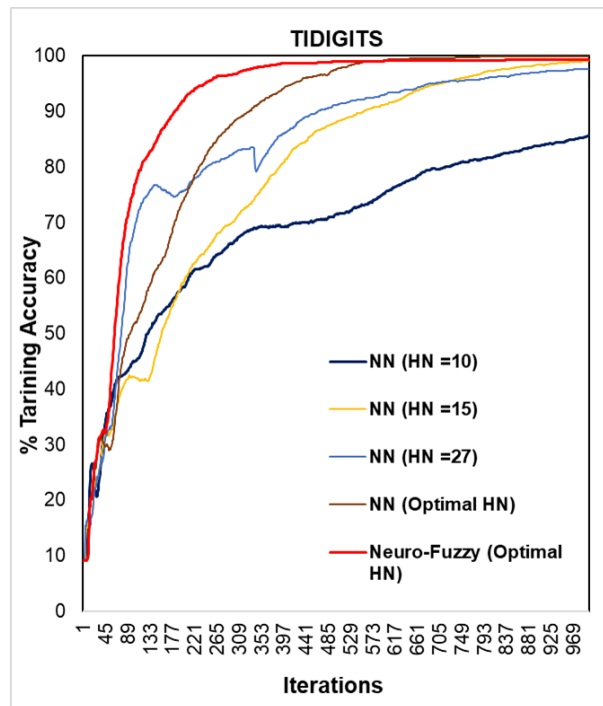
Table 2. Designed parameters for BPNN

Parameter	SMND	TIDIGITS	VD
Input Neurons	300	300	300
Number of HL	1	1	1
Output neurons	10	11	12
Iterations	1000	1000	1000

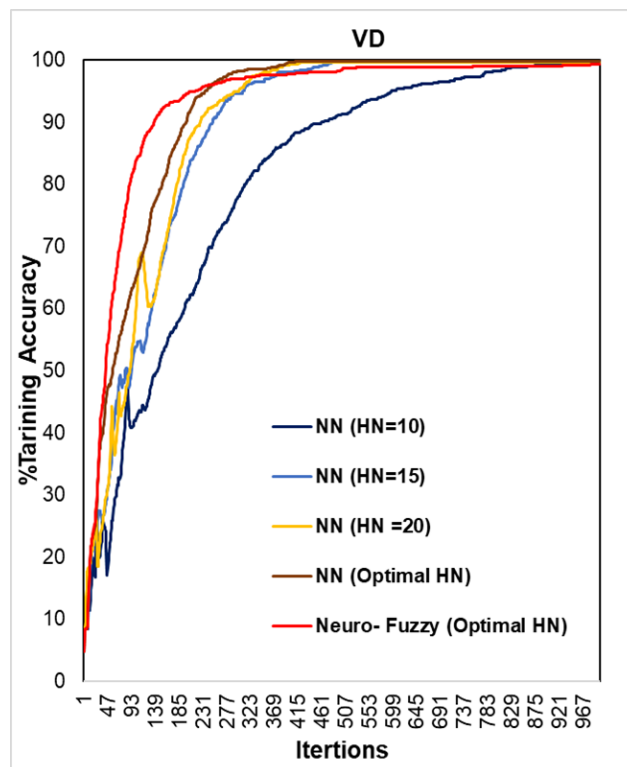
Increasing the number of hidden neurons improves the performance of the neural network. Fig. 3 shows the effect of varying number of hidden neurons on training accuracy of BPNN. The network is trained for 1000 iterations for all the three datasets. Conventional BPNN is trained for different number of hidden neurons based on experimentation. The fuzzy controller finds optimal value of hidden neurons for proposed Neuro-Fuzzy classifier. The plot shows that in terms of training accuracy, the proposed hidden neurons tuning through fuzzy controller converges faster and outperforms conventional network during training. For all the three datasets, maximum accuracy is achieved in the least number of iterations. Thus, the fuzzy controller provides a optimal value of hidden neurons that gives higher training accuracy while converging faster than conventional BPNN network. The proposed Neuro-Fuzzy approach incorporates hidden neurons tuning by the fuzzy controller to avoid trial and error runs and guarantees faster convergence. The optimal hidden neurons value estimated by the fuzzy controller for SMND, and TIDIGITS is 37 each. While it is 29 for VD. BPNN is trained with these values of hidden neurons to test the effectiveness of the proposed approach. In case of SMND, the training accuracy of the proposed technique is 94.32% while it is 85.30% for the later on.



(a)



b)



(c)

Figure 3. Effect of hidden neurons on training accuracy. (a) SMND dataset (b) TIDIGITD dataset (c) VD dataset.

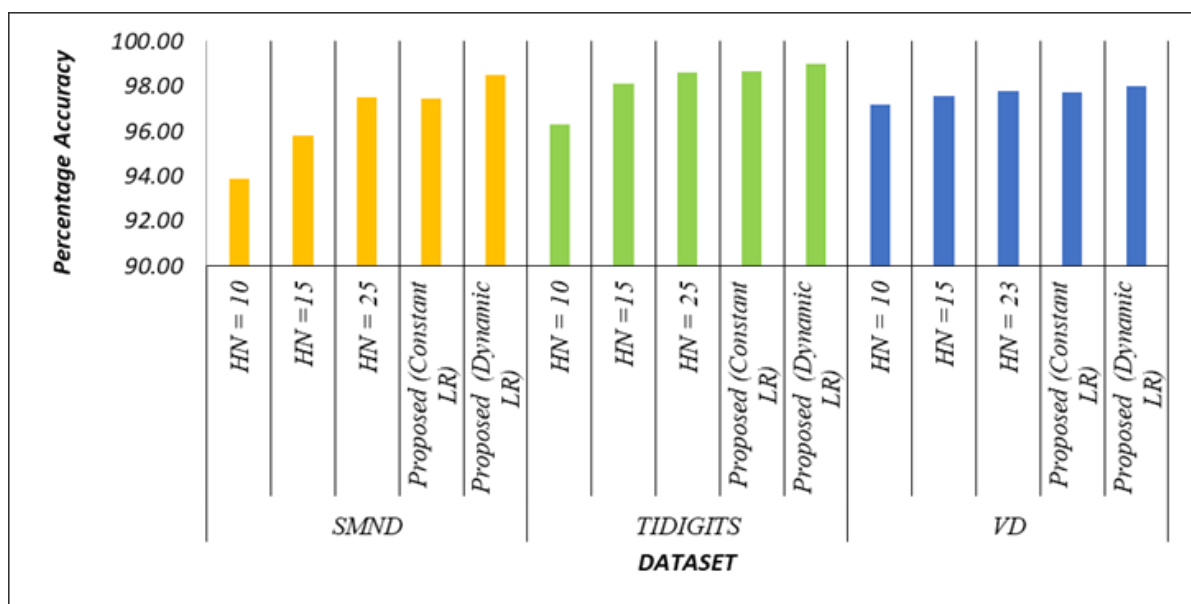
Table 4 lists some of the state of the art hidden neurons fixation criteria implemented for TIDIGITS dataset. i is the number of inputs, o is the number of outputs and T is the number of training samples. 's' is a constant between 0 and 10. Minimum MSE is attained using proposed fuzzy controller. Trained network details generated by Neuro-Fuzzy classifier are used for classifying speech data. The size of the testing dataset is mentioned in Table 1. Performance evaluation of the Neuro-Fuzzy classifier is presented in Fig. 4. The performance criteria used are classification accuracy and precision. Accuracy is the proportion of estimates that are accurate. Precision is the percentage of positive predictions that are accurate. Classification results show that Neuro-Fuzzy classifier produces higher accuracy as well as precision than the conventional neural network classifier for SMND and TIDIGITS datasets. The performance of both the classifiers is comparable for VD dataset. Thus, Neuro-Fuzzy classifier allows training the network with fewer iterations improving the testing performance. HN tuning by fuzzy controller results in training accuracy that moves up faster and smoother than the conventional approach. For all the three datasets, maximum accuracy is achieved in the least number of iterations. Thus, the fuzzy controller provides a near-optimal value of HN that gives higher training accuracy while converging faster than conventional BPNN network. Another advantage is fast processing as it avoids experimentations to fix the optimal value of HN. Highest accuracy provided by the proposed classifier is 99.03%. Highest precision achieved is 94.73%. Table 4 represents the comparative results of the proposed technique with existing classification techniques. An accuracy percentage as high as 99.03% for TIDIGITS compared to other existing neural network approaches implemented for speech recognition. The proposed technique gives a better value of MSE in comparison with other existing models. Experimental findings show that the technique provides better classification results for speech recognition applications when opposed to other approaches.

Table 3. Performance analysis of various approaches for fixation of hidden neurons for TIDIGITS dataset.

Sr. No.	Various methods	Hidden neurons fixation criteria	MSE
1	Gnana Sheela et al., 2013 [9]	$HN = (4i^2 + 3)/(i^2 - 8)$	0.00230
2	Madhiarasan et al., 2017 [14]	$HN = (4i - 2)/(i - 3)$	0.00230
3	Gevaert et al., 2010 [15]	$HN = \frac{T}{5(i + o)}$	0.0257
4	Zhang et al., 2019 [16]	$i = o$	0.000547
5	Qian et al., 2013 [17]	$HN = \sqrt{i} \cdot s$	0.000395
6	Al-Alaoui et al., 2008 [18]	$HN = \sqrt{i * o}$	0.00020
7	Proposed Approach	Neuro-Fuzzy (Fuzzy controller)	0.00013

Table 4. Comparative performance of state of the art techniques and proposed technique

Sr. No.	Various methods	Classifier	Dataset	Testing accuracy
1	Geveart et al., 2010 [15]	BPNN	English numerals dataset	85%
2	Das et al., 2012 [19]	Radial basis function neural network	Isolated words speech dataset	85%
3	Anoop et al., 2018 [20]	BPNN	TIDIGITS	98.20%
4	Alex et al., 2018 [21]	BPNN	TIDIGITS	95%
5	Kaur et al., 2018 [22]	BPNN	TIDIGITS	97%
6	Mustafa et al., 2019 [23]	Dynamic multilayer perceptron	TIDIGITS	96.94%
7	Proposed approach	Neuro-Fuzzy	TIDIGITS	99.03%



(a)

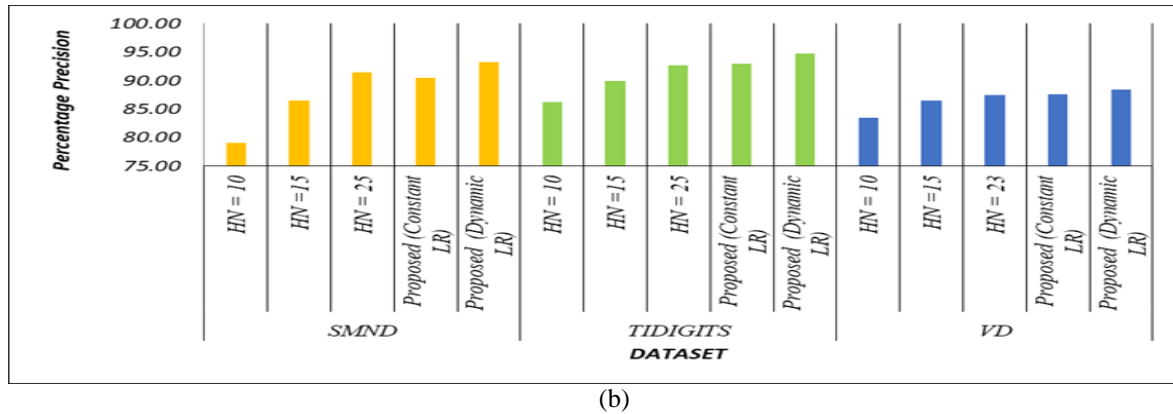


Figure 4. Performance evaluation of proposed Neuro-Fuzzy classifier for speech recognition. (a) accuracy plot (b) precision plot.

CONCLUSION

This research paper introduces a hybrid classifier capable of automatically tuning the number of hidden neurons to an optimal value. It has been shown that the network converges fast with a considerable reduction in MSE and number of iterations. The highest training accuracy of 99.27% with MSE value as low as 0.00013 is achieved by the hybrid Neuro-Fuzzy classifier. This ensures enhanced testing accuracy of 99.03% and precision of 94.73% for speech recognition. For qualitative analysis, we have plotted the convergence curves for conventional BPNN and proposed hybrid algorithm. The curves showed that the technique provides faster convergence while maintaining testing accuracy for speech datasets. Though the algorithm is applied for speech recognition in this paper, it may be effectively used in other soft computing applications. Further, the research work presented in this paper can be extended to test on other types of neural networks.

REFERENCES

- Goh, TC Anthony, "Back-propagation neural networks for modeling complex systems." *Artificial intelligence in engineering* vol. 9, no. 3, 1995, pp. 143-151.
- W. D. Randall and T. R. Martinez. "The need for small learning rates on large problems." In *IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No. 01CH37222)*, vol. 1, pp. 115-119. IEEE, 2001.
- Li, Y.J., "A method to directly estimate the number of the hidden neurons in the feedforward neural networks", *J. Comput.*, vol. 22, no. 11, pp. 1204-1208, 1999.
- H. C. Yuan, F. L. Xiong, and X. Y. Huai. "A method for estimating the number of hidden neurons in feed-forward neural networks based on information entropy." *Computers and Electronics in Agriculture* 40, no. 1-3, 2003, pp. 57-64.
- Z. Boger, and H. Guterman. "Knowledge extraction from artificial neural network models." In *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, vol. 4, pp. 3030-3035. IEEE, 1997.
- M.J.A Berry, & G. Linoff, "Data Mining Techniques", NY: John Wiley & Sons, 1997.
- Blum, *Neural Networks in C++*, NY: Wiley, 1992.
- Liu, Yinyin., Janusz, A. Starzyk, & Z. Zhen, "Optimizing number of hidden neurons in neural network", *EeC*. vol. 1, no.1 pp. 6, 2007.
- K. Sheela, Gnana, and N. D. Subramaniam, "Review on methods to fix number of hidden neurons in neural networks." *Mathematical Problems in Engineering* 2013.
- F. S. Panchal, & M. Panchal, "Review on methods of selecting number of hidden nodes in artificial neural network", *International Journal of Computer Science and Mobile Computing*, vol. 3, no. 11, pp. 455-46, 2014.
- H. Kondhalkar, P. Mukherji, "A database of Marathi numerals for speech data mining. *International Journal of Advanced Research in Science and Engineering*", vol. 6, no. 10, pp. 395-399, 2017.
- ece.ucsb.edu/Faculty/Rabiner/ece259/speechrecognitioncourse.html.
- J. Hillenbrand, L. A. Getty, M.J. Clark & K. Wheeler, K., "Acoustic characteristics of American English vowels", *The Journal of the Acoustical Society of America*, vol. 97, no. 5, pp. 3099-3111, 1995.
- M. Madhwaran, & S. N. Deepa., "Comparative analysis of hidden neurons in multi-layer perceptron neural networks for wind speed forecasting", *Artificial Intelligence Review*, vol. 48, no. 4, pp. 449-471, 2017.

14. W. Gevaert, G. Tsenov & V. Mladenov, "Neural networks used for speech recognition", *Journal of Automatic control*, vol. 20, no. 1, pp. 1-7, 2010.
15. S. Zang, W. W. Ng, J. Zhang, C. D. Nugent, N. Irvine & T. Wang, "Evaluation of radial basis function neural network minimizing L-Gem for sensor-based activity recognition", *Journal of Ambient Intelligence and Humanized Computing*, pp. 1-11, 2019.
16. G. Qian, & H. Yong, "Forecasting the rural per capita living consumption based on Matlab BP neural network", *International Journal of Business and Social Science*, vol. 4, no. 17, pp. 131-137, 2013.
17. M. A. Al-Alaoui, L. Al-Kanj, J. Azar, & E. Yaacoub, "Speech recognition using artificial neural networks and hidden Markov models. *IEEE Multidisciplinary engineering education Magazine*", vol. 3, no. 3, pp. 77-86, 2008.
18. B. P. Das, & R. Parekh, "Recognition of isolated words using features based on LPC, MFCC, ZCR and STE, with neural network classifiers", *International Journal of Modern Engineering Research*, vol. 2, no. 3, pp. 854-858, 2012.
19. V. Anoop, P. V. Rao, & S. Aruna, "An Effective Speech Emotion Recognition Using Artificial Neural Networks", *Advances in Intelligent Systems and Computing*, vol. 628, pp. 393-401, 2018.
20. J. S. R. Alex, A. Das, S. A. Kodgule, & N. Vekatesan, "A comparative study of isolated word recognizer using SVM and wavenet. Integrated speaker and speech recognition", In *proceedings of the Computational Signal Processing and Analysis*. Singapore, 2018.
21. G. Kaur, M. Srivastava, & A. Kumar, "Integrated speaker and speech recognition for wheel chair movement using artificial intelligence. *Informatica*", vol. 42, no.4, pp. 587-594, 2018.
22. M. K. Mustafa, T. Allen, & K. Appiah, "A comparative review of dynamic neural networks and hidden Markov model methods for mobile on-device speech recognition", *Neural Computing and applications*, vol. 31, no.2, pp. 891-899, 2019.
23. R. Hecht-Nielsen, "Theory of the backpropagation neural network. *Neural networks for perception*", 65-93, 1992.
24. S. I. Amari, "Backpropagation and stochastic gradient descent method. *Neurocomputing*", 5(4-5), 185-196, 1993.
25. X. Yao, "Evolving artificial neural networks," in *Proceedings of the IEEE*, vol. 87, no. 9, pp.1423-1447, 1999.
26. Z. Amiri, H. Hassanpour, N. M. Khan and M. H. M. Khan, "Improving the performance of multilayer Backpropagation neural networks with adaptive learning rate," in *2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD)*, IEEE, 2018, pp. 1-4.
27. D.R. Wilson and T. R. Martinez, "The need for small learning rates on large problems," in *IJCNN'01, International Joint Conference on Neural Networks. Proceedings (Cat. No. 01CH37222)*, IEEE, vol. 1, pp. 115-119, 2001.
28. M. W. Lam, S. Hu, X. Xie, S. Liu, J. Yu, R. Su, X. Liu and H. Meng, "Gaussian process neural networks for speech recognition," in *Interspeech*, 2018, pp. 1778-1782.
Rakitińskaia and A. Engelbrecht, "Measuring saturation in neural networks," in *2015 IEEE Symposium Series on Computational Intelligence*, IEEE, 2015, pp. 1423-1430.
29. Z. Amiri, H. Hassanpour, N. M. Khan and M. H. M. Khan, "Improving the performance of multilayer Backpropagation neural networks with adaptive learning rate," in *2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD)*, IEEE, 2018, pp. 1-4.