International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-6, August 2019

ANFIS for Tamil Phoneme Classification

Laxmi Sree B.R., Vijaya M.S.

Abstract**:** ***Phoneme recognition is an intricate problem lying under non-linear systems. Most research in this area revolve around try to model the pattern of features observed in the speech spectra with the use of Hidden Markov Models (HMM), various types of neural networks like deep recurrent neural networks, time delay neural networks, etc. for efficient phoneme recognition. In this paper, we study the effectiveness of the hybrid architecture, the Adaptive Neuro-Fuzzy Inference System (ANFIS) for capturing the spectral features of the speech signal to handle the problem of Phoneme Recognition. In spite of a wide range of research in this field, here we examine the power of ANFIS for least explored Tamil phoneme recognition problem. The experimental results have shown the ability of the model to learn the patterns associated with various phonetic classes, indicated with recognition improvement in terms of accuracy to its counterparts.***

Keywords **:** ***Adaptive Neuro-Fuzzy Inference System, Phoneme Recognition, Speech Recognition, Tamil Phoneme Classification***

I. INTRODUCTION

Speech is one of the most powerful sources that trigger the hearing process and helps improve one's cognitive capabilities. It is the era where human machine interaction is growing faster, and in an unimaginable way. Speech recognition and voice recognition technologies help a lot in the development of this arena. Speech recognition play a major role in resolving the barriers of the common people in using the technology, as it acts a bridge that add an artificial cognitive hearing capability to the gadgets around him. Even the illiterate or people with visual impairments can use this technology to cross their barriers in use most powerful technological advancements floating around him. Technological advancements can be made to reach people in each nook and cranny of the world by presenting the core services through their native speech interface that makes them more comfortable.

Automatic speech recognition (ASR) has turned out to be one of the powerful interfaces to communicate with devices. Various studies have been undergone in this area to facilitate automatic speech recognition of continuous speech but only a comparably least concentrate on Tamil continuous speech.

Revised Manuscript Received on August 30, 2019.
* Correspondence Author
Laxmi Sree B.R.*, Department of Computer Science, Dr. G.R. Damodaran College of Science, Coimbatore, India. Email: viporala@yahoo.com
Vijaya M.S., dDepartment of Computer Science, PSGR Krishnammal College for Women, Coimbatore, India. Email: msvijaya@psgrkcw.ac.in

Most research in speech recognition use Hidden Markov Models, Support Vector Machines and few recently use various types of neural networks [1]-[4]. Success of automatic speech recognition lies in the correctness of mapping the speech signals comprising the spoken sentences to the respective phonemes, which is then fed to the language model to identify the spoken words. This involves several steps that include feature extraction from raw data, segmentation of continuous speech into smaller units like syllables or phonemes, classifying the phonemes and identifying the spoken words. With an objective to support human machine interaction for Tamil speaking community, the article focuses on a study to develop a Tamil phoneme recognition model using Adaptive neuro-fuzzy inference system, a machine learning technique to build an artificial cognitive knowledge base that helps the machine to identify the spoken Tamil phonetic units. ANFIS is an adaptive network that combines the pros of the neural network that dynamically learns itself from the data available and fuzzy logic that is capable of building decision rules from uncertain information. This paper is further organized as follows; Section 2 discusses the previous studies on fuzzy logic and ANFIS whose result motivated to build a Tamil phoneme recognition model using ANFIS. Section 3 details on the fuzzy logic system, section 4 on the architecture and working of ANFIS, which is followed by ANFIS model for Tamil phoneme recognition in section 5, Experimental results and discussion in section 6 and Conclusion in section 7.

II. RELATED WORK

LAS- Listen, Attend and Spell , a neural network model is developed to transcribe the speech utterance to characters or graphemes, rather than encoding to phonemes. The architecture of LAS is composed of two components namely, the listener and speller [5]. The listener is built as a pyramid structure using Bidirectional Long Short Term Memory Recurrent Neural Network that is trained to transform the input speech signals into high level feature representations, which is fed to the speller of the architecture to identify the corresponding graphemes. The speller is designed as a attention based LSTM transducer, which computes the probability distribution of next possible character based on the previously seen characters. The model has proved to be more efficient even in absence of an language model and conditional independence assumptions on the input speech.

An analysis of a hybrid system that combines the Convolution Neural Networks (CNN) with Connectionist Temporal Classification (CTC) approach is used to identify the phonemes in an unsegmented speech signal [6].

The system trains itself with the capability of CNN to reduce the spectral variations observed in various instances of the speech signal and to cognitively model the correlations observed in the acoustic features of the speech spectrum; which is further a tied with CTC to label the unsegmented signal. The CNN with CTC system with deeper architecture and better regularization has rolled out with better classifications. A direct approach that combines a sequence of processes acoustic modeling, language modeling and sequence prediction in continuous speech recognition is developed using Recurrent Neural Networks (RNN) [7]. The character level sequence prediction is performed directly by using an attention mechanism that automatically learns to align input features with the desired character sequence and choosing the relevant frames of the characters. It finally integrates an n-gram language model to provide better decoding results. The relationship between the raw speech signal and the phoneme units in the continuous speech are modeled with 2-layer convolution neural network (CNN) [8]. This models the spectral information of about 2-4ms sub-segmental speech. The study has shown that the CNN based recognition work comparatively better to the ASR systems built with short-term spectrum even in noisy environment. [9] proposed to integrate different features like mel-filter bank energies, cepstral co-efficients, their first and second order derivatives into one entity in the form quaternion. The sequence-to-sequence mapping as in connectionist temporal classification is performed by integrating the multi-featured quaternions with the CNNs forming quaternion-valued CNNs (QCNN). It has been observed the QCNNs outperformed the traditional CNNs and other real-valued counterpart with significantly fewer parameters and robust for noisy speech. ANFIS has been successfully used for many of the classification problems in the field of medicine, speech recognition, agriculture, production industry and more [10]-[12]. A noise robust speech recognition system, Wavelet Packet Adaptive Fuzzy Network Inference System (WPSNFIS) is developed in [13], which comprises of two layers. The first layer, Wavelet packet layer performs an adaptive feature extraction using wavelet decomposition and wavelet entropy. The next layer uses ANFIS for the classification task. Its classification accuracy of about 92% has proved its efficiency to speech recognition. A novel neuro-fuzzy classification model has been developed by building membership matrix between the features and the classes to explore the feature-wise membership to different classes [14]. This method has proved to be best performing in terms of various measures including misclassifications, classification accuracy, kappa index and β index even for less training data. High performance of the model is achieved with a complexity trade-off. An Adaptive Fuzzy Inference Neural Network (AFINN) which self-constructs the rules with appropriate parameter adjusting developed and tested for few non-linear systems, namely formula approximation, speech recognition and car evaluation [15]. AFINN's simple structure is capable enough to overcome the shortcoming of conventional fuzzy neural network and construct a classification model with optimal number of rules, elements of input and model parameters with a good accuracy record.

III. FUZZY LOGIC SYSTEM

Fuzzy logic system is a control system that takes the degrees of n real valued inputs and produces output based on the input state and the rate of change of those states. It uses an approximate reasoning methodology which uses linguistic terms to define the inputs and generate outputs even for the ambiguous and noisy data. The fuzzy logic system is composed of three components namely, fuzzifier, controller and defuzzifier.

A. Fuzzifier

The input feature vectors are given as input to the fuzzifier which transforms the input to linguistic variables like very low, low, medium, high and very high. The physical values of inputs are mapped to a normalized fuzzy subset of values defined using membership functions. The membership function can be of any type like sigmoidal, trapezoidal, triangular or others that define the probability of the input variable to be in a particular state.

B. Controller

The controller comprises of the knowledge base and inference engine. The knowledge base is built during the supervised learning process by generating the fuzzy rules from the training dataset, once the membership functions are defined for the linguistic variables in the knowledge base. With the model built, the inference engine in the controller uses the knowledge base to classify the new feature vector given as input.

C. Defuzzifier

This performs the reverse function of fuzzifier. The fuzzy output received from the controller is transformed to numerical form of physical values.

For example, consider the problem of designing a fuzzy controller to control the temperature of room air conditioner based on the current room temperature and number of persons in the room. The linguistic variables for room temperature can be defined as low, medium, high and for number of persons as less, moderate and more. The membership functions of room temperature are defined to be triangular functions as shown in Fig. 1.

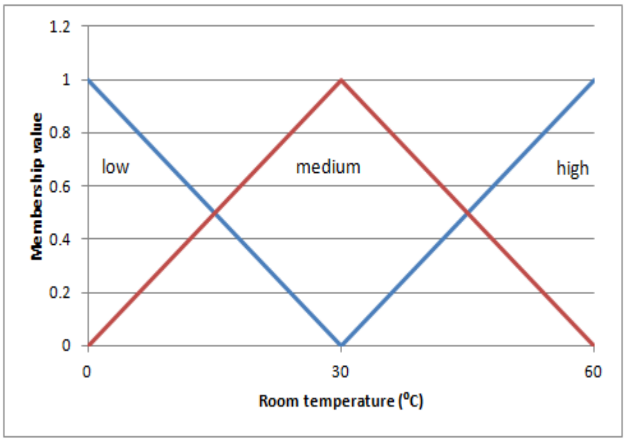

Fig. 1. Triangular membership functions defining the linguistic terms of room temperature variable.

International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-6, August 2019

The rules in the knowledge base take IF….THEN form which looks as follows:

IF room temperature is low AND number of persons is less THEN set AC temperature as low

IF room temperature is low AND number of persons is more THEN set AC temperature as medium

IF room temperature is medium AND number of persons is more THEN set AC temperature as high

IF room temperature is high AND number of persons is moderate THEN set AC temperature as high

and so on.

IV. ADAPTIVE NEURO-FUZZY INFERENCE SYSTEM (ANFIS)

Adaptive Neuro-Fuzzy Inference System is a combination of Adaptive Neural Networks and Fuzzy logic which takes the advantage of soft computing techniques of neural networks and the ability of Fuzzy logic to transform human knowledge into quantitative rules [16]. The type-3 ANFIS with an adaptive network is a feed forward network that has the capability to learn itself. The type-3 ANFIS [17] is a five layer feed forward network whose structure with two inputs, one output and nine rules is shown in the Fig. 2, where the square node denote the adaptive nodes, that are capable of learning and the circle nodes are fixed nodes. The rules of the network are Takagi and Sugeno type. The consequent part of the rules' generated are built as a linear combination of the inputs and a constant term. The weighted average of each rule's output is given as the final output of the network. The type-3 ANFIS architecture can be explained as follows:

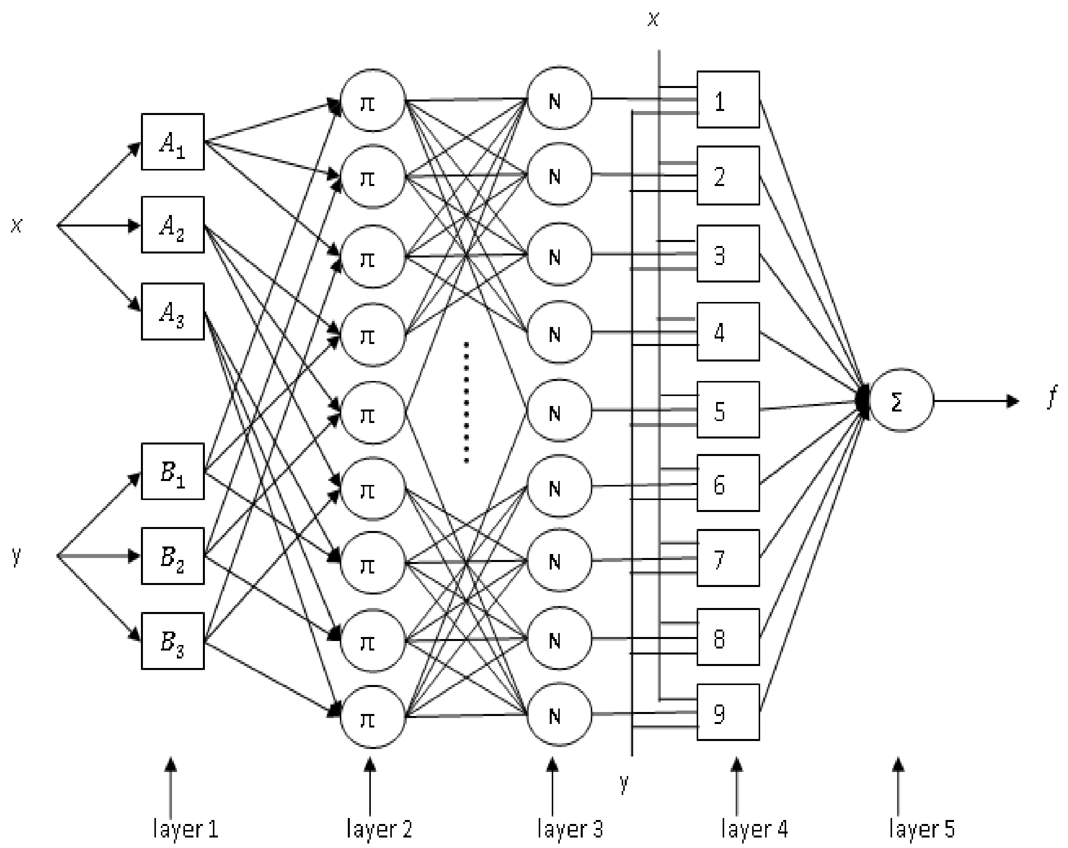

Fig. 2. Type-3 ANFIS

Layer 1: This layer is a fuzzification layer, where the inputs are fuzzified to represent the linguistic terms of the system. The nodes of this layer are adaptive and are represented by some continuous piecewise differentiable functions like trapezoidal, triangular or bell-shaped functions. Each node of this layer is defined with a membership function and is represented as follows,

$$O_i^1 = \mu A_i(x) \tag{1}$$

where x is the input for the node i, and A_i is a function representing the linguistic term associated with the node. For a bell-shaped membership function, the membership values usually lies between 0 and 1 and is given as follows,

$$\mu A_i = \frac{1}{1+\left[\left(\frac{x-c_i}{a_i}\right)^2\right]^{b_i}} \tag{2}$$

where the parameter defining the bell-shape a_i, b_i and c_i forms the parameter set. These parameters are thereby referred as premises parameters, as they help defining the premises part of the rule set.

Layer 2: Each node in this layer is a circle node whose function is to produce the product of all incoming signals from the previous layer.

The output of the nodes in this layer indicates the firing strength of the rule, which is represented as,

$$w_i = \mu A_i(x) \times \mu B_i(y) \quad (3)$$

with $i=1,2$ for two input network, $\mu A_i(x)$ and $\mu B_i(y)$ denoting the membership function for the respective inputs.

Layer 3: This layer represented with nodes labelled N, gives the ratio of the firing strength of the i^{th} rule (w_i) to the sum of firing strengths of all rules. For the given nine rule network, this ratio is calculated as follows,

$$\overline{w_i} = \frac{w_i}{\sum_{i=1}^{9} w_i} \quad (4)$$

Layer 4: Each node in this layer is represented as a combination of the output of layer 3 and linear function of inputs and given by

$$O_i^4 = \overline{w_i}(p_i x + q_i y + r_i) \quad (5)$$

where ($\overline{w_i}$) is the output of previous layer and p_i, q_i and r_i are the consequent parameters, the parameters of layer 4.

Layer 5: This layer is the output layer, which provides the summation of all incoming signals and is given by,

$$O_1^5 = \sum_i O_i^4 \quad (6)$$

With this structure, the ANFIS model follows a hybrid learning rule, which employs two passes – the forward pass and the backward pass. In the forward pass, the network learns from the functional signals that moves forward through the layers of the network. In the backward pass, the least square error is calculated and is back propagated to each layer through gradient descent.

Back propagation in ANFIS

During the training process, in the backward pass, the error observed in the output layer is back propagated to the previous layers to fine tune the premise parameters $\{a_i, b_i, c_i\}$ that defines the shape of the membership functions [18]. The error at the output layer is calculated as a squared error follows:

$$E_p = \sum_{k=1}^{N_L}(d_k - X_{k,p}^L)^2 \quad (7)$$

where d_k is the k^{th} component of the desired output, vector $X_{k,p}^L$ is the k^{th} component of the actual output of ANFIS in forward pass for p^{th} input data vector, N_L denotes the number of nodes in layer L. The error starts propagating backward through each and every node of layer 4 of the network as a derivative of square error which is given by,

$$\varepsilon_{L,j} = \frac{\partial E_p}{\partial_{j,p}^L} = -2(d_j - X_{j,p}^L) \quad (8)$$

$\forall j$ node in layer L. Further, the internal nodes lying in the other layers *(l)* of ANFIS are updated by using the chain rule as follows,

$$\frac{\partial E_p}{\partial x_{l,i}} = \sum_{u=1}^{N_{l+1}} \frac{\partial E_p}{\partial x_{u,p}^{l+1}} \frac{\partial x_{u,p}^{l+1}}{\partial x_{l,i}} \quad (9)$$

With the internal node updated with the error as in the above equation, any parameter *y* of some set of nodes, *S* in the networks can be updated as follows,

$$\frac{\partial E_p}{\partial y} = \sum_{z \in S} \frac{\partial E_p}{\partial z} \frac{\partial z}{\partial y} \quad (10)$$

Thus the overall error of parameter y is given as a summation of error contributed by each input data and the gradient descent for the parameter y is given as

$$\Delta y = -\gamma\, \partial E / \partial y \quad (11)$$

with γ being the learning rate.

V. ANFIS MODEL FOR TAMIL PHONEME RECOGNITION

In this paper, an experimental study on ANFIS for building a model for the problem of Tamil phoneme recognition is undergone. In this study, database “Kazhangiyam”, developed in our earlier work is used for building a phoneme recognition model. The dataset used here is a collection of data points, where each data point represents the DWT features of a phoneme, which has been segmented from the continuous Tamil speech using a graph-cut based segmentation algorithm [19]. Each phoneme is defined as a data point with 90 DWT features. A subset of 3040 data points spanning all the 39 classes of the dataset with 80 samples in each class is used here for training ANFIS, where as the test dataset is a subset of 1520 data points contributed with 40 samples from each class.

The complete flow diagram showing the steps in building ANFIS based acoustic model is shown in Fig. 3. Initially Linear Discriminant Analysis (LDA) is applied on the dataset to reduce the dimensionality of the feature space. LDA is a powerful feature extraction technique that linearly transforms

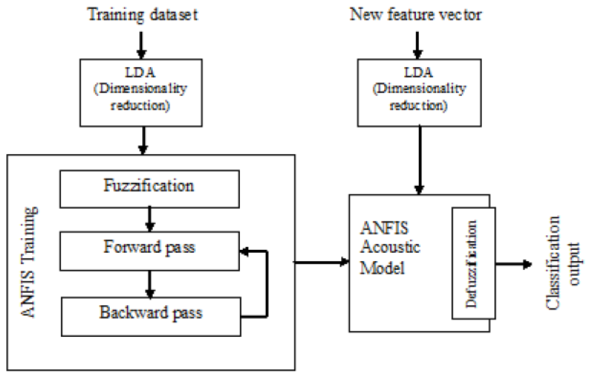

Fig. 3. Flow diagram to build ANFIS based acoustic model for Tamil phoneme recognition

The features from the original space into a new space with lower dimensionality. The resulting feature space has nine features representing the phonetic units. As a first step to build the ANFIS model, each input variable is fuzzified with three linguistic terms namely low, medium and high and represented by Gaussian functions. The layer 1 of our ANFIS model has 27 nodes representing the linguistic units of nine features in the new space with each linguistic term represented as Gaussian function. The model is then trained with the training dataset using two passes, where the ANFIS learns by propagating the input till the output layer and fine tunes the parameters in the backward pass using gradient based back-propagation technique to propel the error observed in the output layer. This learning procedure of forward and backward pass is repeated to few hundred epochs to train the network. The experimental results of training the ANFIS for Kazhangiyam phoneme dataset are discussed in the following section.

VI. EXPERIMENTAL RESULTS AND DISCUSSION

The phonemes in Tamil language, grouped under various broad categories are listed in Table I.

The category vowel is formed with twelve phonetic classes (short vowels, long vowels and dipthongs), nasals with seven phonetic classes, stops with eight and the remaining eleven classes grouped under others (fricatives, glides and grantha) category. The distribution of data in the dataset for various categories is shown in Fig. 4 and the distribution of samples from various classes within each category is visualized in Fig 5.

Table- I: Categories of Phonemes in Tamil Language

Broad Categories	**Phonetic Classes**
Vowels	/a/,/aa/,/i/,/ii/,/u/,/uu/,/e/,/ee/,/ai/,/o/,/oo/,/au/
Nasals	/m/,/n/,/N/,/nn/,/nj/,/gh/,/ng/
Stops	/p/,/b/,/t/./d/,/th/,/dh/,/k/,/g/
Others	/s/,/zh/,/ch/,/sh/,/ei/,/r/,/rr/,/L/,/j/,/v/,/l/

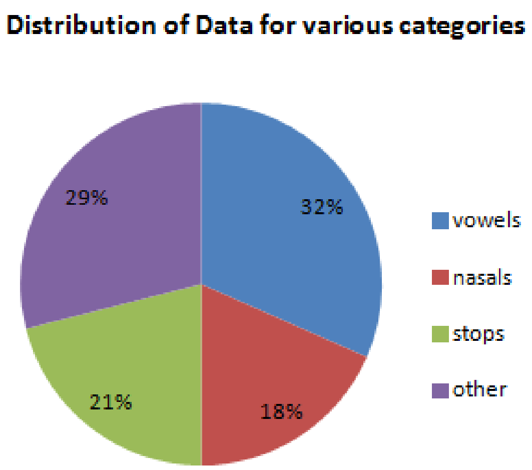

Fig. 4. Comparison of number of rules generated and time taken by ANFIS with respect to dataset size

The training set and the test data for developing the ANFIS acoustic model has been built in 2:1 ratio. LDA is applied on the dataset representing 90 features of phoneme to reduce the feature space dimensionality, which is then fed to the ANFIS learning module to train the model. The built acoustic model is tested with the test set and the performance is studied.

The accuracy of the ANFIS model built has been analyzed based on the various standard measures in the literature: precision, recall, F-measure and classification accuracy on the broad classes of phonemes namely, vowels, nasals, stops and others. The measures used to study the efficiency of the model built are defined below.

Precision defines the proportion that has been correctly identified as positives to the total number of identified positive classifications and is given as follows,

$$\text{Precision} = \frac{\text{True positives}}{\text{True positives+False positives}} \quad (12)$$

Recall defines the proportion of actual positives that was identified correctly as positives and is given as follows,

$$\text{Recall} = \frac{\text{True positives}}{\text{True positives+False Negatives}} \quad (13)$$

F-measure is a score that is defined to be the harmonic mean of both precision and recall and is given by,

$$\text{F} - \text{measure} = 2.\frac{\text{Precision.Recall}}{\text{Precision+Recall}} \quad (14)$$

The precision, recall and F-measure observed for various categories of phonemes is portrayed in Table II. It is observed that the efficiency of the ANFIS acoustic model improves with the number of samples in the training dataset, but it can be observed that the rate of improvement in precision, recall and F-measure is minimal when compared to the rate of increase in number of samples (Fig. 6). Also, the number of rules generated by the ANFIS model and the execution time is observed to be increasing at greater rates with increase in the number of samples (Table III). The number of rules generated is 121, 384, 665 and 956, where as the execution time taken by ANFIS to build the model from the training dataset is observed as 153, 198, 285 and 360 minutes respectively for a dataset of size 780, 1560, 2340 and 3120 samples. From Fig. 5, it can also be seen that the precision, recall and F- measures increase rate is diminishing with the greater overheads of knowledge base size (number of rules) and execution time. This greater increase in the number of rules generated is due to the increase in dataset size by adding more number of samples from different speakers. This shows the higher variability in the speech characteristics of the speakers. The experiments in building the ANFIS model for varied dataset sizes show that there is a linear increase in the number of rules generated (Table III and Fig. 7) with the increase number of samples in training. This shows the high variability in the features representing the phonetic units of the language, reflecting the characteristics of multiple speakers and the effect of co-occurring phonemes because the numbers of samples in the datasets were increased by including samples from more speakers. The time taken is also observed to be high for training ANFIS even for a smaller dataset of size 780. The study also shows that the increase in the number of samples does not reflect the model efficiency much as expected in terms of accuracy rather it increases the time overhead in training.

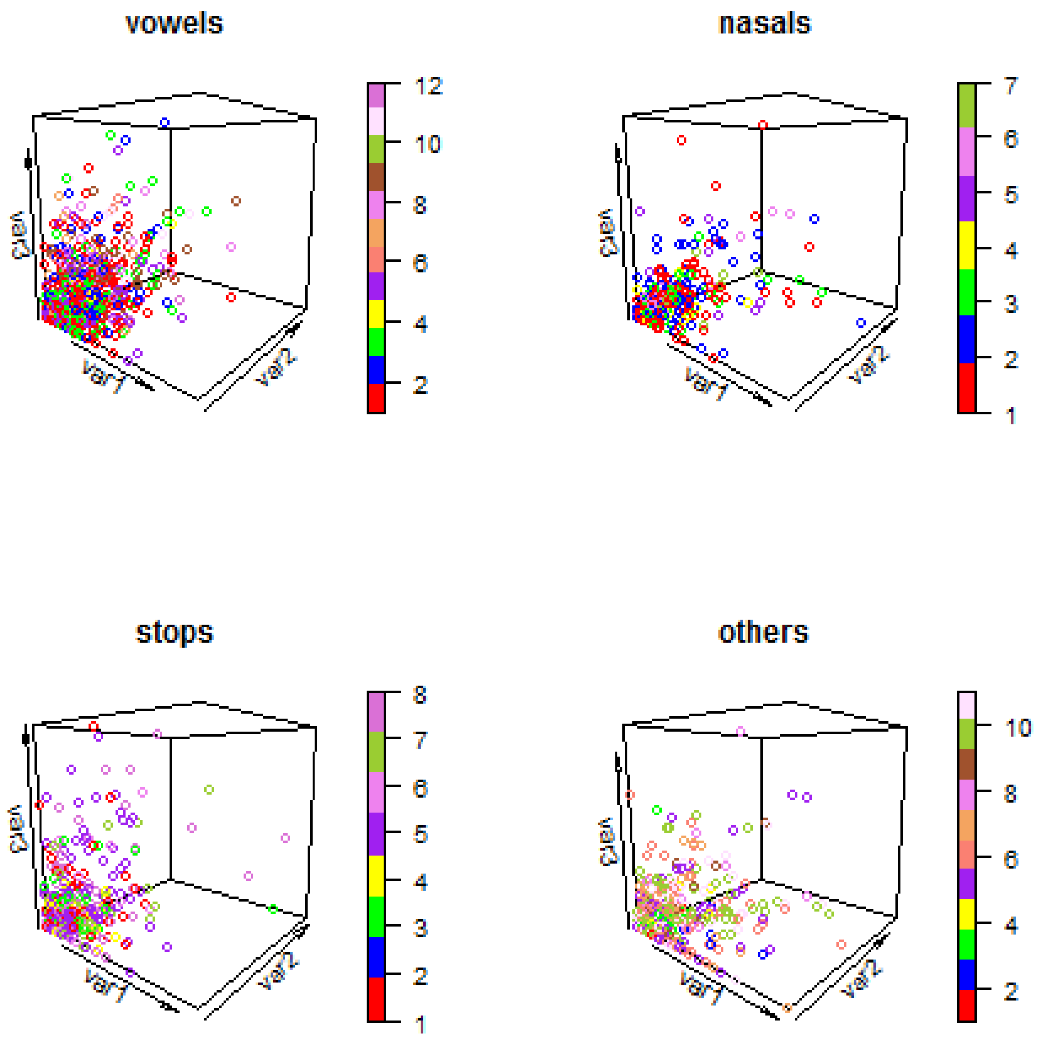

Fig. 5. Distribution of Samples from Various Classes in each Category

Table- II: Average Precision, Recall And F-Measure observed for ANFIS Models built for Phoneme Dataset of Various Size

Number of Samples in Training dataset	**Average Precision**	**Average Recall**	**F-measure**
780	0.34	0.45	0.3873
1560	0.38	0.51	0.4355
2340	0.58	0.54	0.5592
3120	0.61	0.56	0.5839

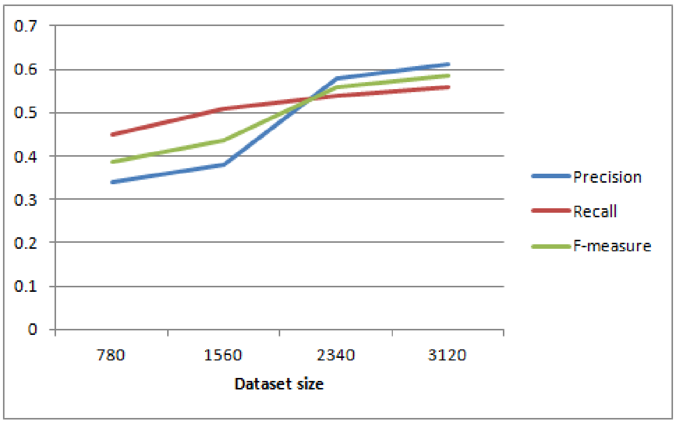

Fig. 6. Comparison of Precision, Recall and F-Measure for ANFIS Model Built with Datasets of Various Size

Table- III: Number of Rules Generated and Time Taken by ANFIS for Various Sizes of Training Dataset

Number of Samples in Training dataset	Number of rules generated	Execution time (in minutes)
780	121	153
1560	384	198
2340	665	285
3120	956	360

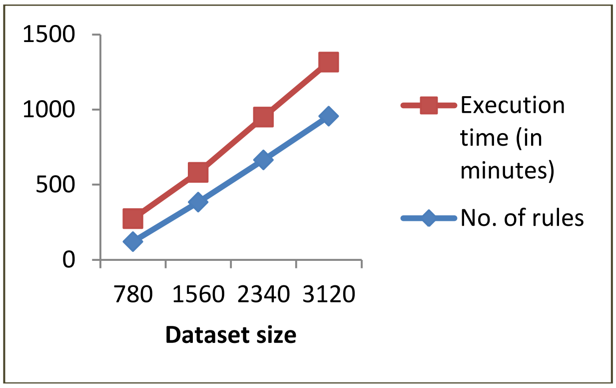

Fig. 7. Comparison of Number of Rules Generated and Time Taken By ANFIS with Respect to Dataset Size

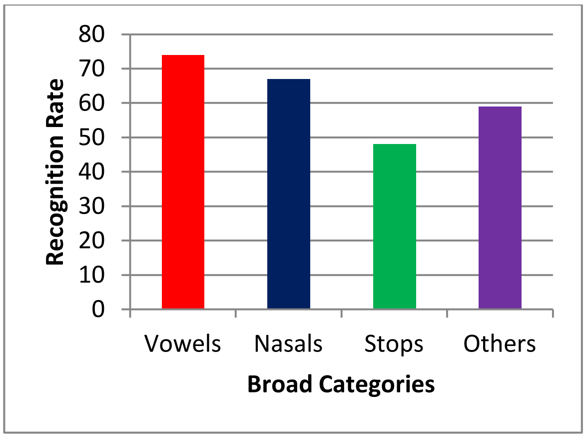

Fig. 8. Comparison of Recognition Rate of ANFIS for Various Categories of Phonemes for a Training Set of Size 3120

The recognition rate with respect to the various categories is compared in the Fig. 8. The recognition rates for the various categories vowels, nasals, stops and others are observed as 74%, 67%, 48% and 59% respectively with an average accuracy of 62% for the ANFIS model trained with dataset of size 3120. The recognition rates for various categories show that the ANFIS model is capable of recognizing the vowels with greater accuracy (74%) when compared to the other type of phonemes in Tamil. During the experiments the phonetic units that span for longer duration were observed to be classified correctly when compared to the phonetic units with shorter duration, thus reflecting greater accuracy for vowels compared to the other categories.

VII. CONCLUSION

This paper studies the ability of the Adaptive Neuro-Fuzzy Inference System to model Tamil phoneme classification problem. Linear discriminant analysis is applied on the data set to reduce the dimensions of the features space, and then phoneme recognition model is built using ANFIS. The results show decent and comparable recognition accuracy to the state of art models available in the literature [20], [21]. The variability in the characteristics of speakers and co-occurrences of phonemes in continuous speech adds more complexity to the problem. The reflection of the complexity of the problem is seen in the complexity of the model built with increase in knowledge base size proportionately to the increase in the training dataset size. Future work will concentrate on building more efficient model for phoneme recognition incorporating improved machine learning techniques capable of handling the variability in the feature space and to reduce the time complexity incurred in building and classification.

REFERENCES

1. Deng, Li, and Xiao Li, "Machine learning paradigms for speech recognition: An overview," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 5, 2013, pp. 1060-1089.
2. Mohamed, Abdel-rahman, George E. Dahl, and Geoffrey Hinton, "Acoustic modeling using deep belief networks," *IEEE Trans. Audio, Speech & Language Processing*, vol. 20, no. 1, 2012, pp. 14-22.
3. Stadermann, Jan, and Gerhard Rigoll, "A hybrid SVM/HMM acoustic modeling approach to automatic speech recognition," *Proc. Int. Conf. on Spoken Language Processing ICSLP# 2004*, Jeju Island, South Korea. 2004.
4. Golowich, Steven E., and Don X. Sun, "A support vector/hidden Markov model approach to phoneme recognition," *ASA Proceedings of the Statistical Computing Section*, 1998.
5. Chan, W., Jaitly, N., Le, Q., & Vinyals, O., Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. *In Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on, IEEE* 2016, pp. 4960-4964.
6. Zhang, Y., Pezeshki, M., Brakel, P., Zhang, S., Bengio, C. L. Y., and Courville, A. "Towards end-to-end speech recognition with deep convolutional neural networks." 2017. *arXiv preprint arXiv:1701.02720*.
7. Bahdanau, Dzmitry, Jan Chorowski, Dmitriy Serdyuk, Philemon Brakel, and Yoshua Bengio, "End-to-end attention-based large vocabulary speech recognition," *In Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on, IEEE*, 2016 pp. 4945-4949.
8. Palaz, Dimitri, and Ronan Collobert, "Analysis of cnn-based speech recognition system using raw speech as input," *Proceedings of INTERSPEECH. No. EPFL-CONF-210029*, 2015.
9. Parcollet T, Zhang Y, Morchid M, Trabelsi C, Linarès G, De Mori R, Bengio Y, "Quaternion Convolutional Neural Networks for End-to-End Automatic Speech Recognition," ar*Xiv preprint arXiv:1806.07789. June 2018.*
10. Ziasabounchi, Negar, and Iman Askerzade, "ANFIS based classification model for heart disease prediction," *International Journal of Electrical & Computer Sciences IJECS-IJENS*, vol. 14, no. 02, 2014, pp. 7-12.
11. Jayachandran, A., & Dhanasekaran, R, "Multi Class Brain Tumor Classification Of MRI Images using Hybrid Structure Descriptor and Fuzzy Logic Based RBF Kernel SVM," *Iranian Journal of Fuzzy Systems*, vol. 14, no. 3, 2017, pp. 41-54.
12. Herman, P. A., Prasad, G., & McGinnity, T. M, "Designing an interval type-2 fuzzy logic system for handling uncertainty effects in brain–computer interface classification of motor imagery induced EEG patterns," *IEEE Transactions on Fuzzy Systems*, vol. 25, no. 1, 2017, pp. 29-42.

13. Avci, Engin, and Zuhtu Hakan Akpolat, "Speech recognition using a wavelet packet adaptive network based fuzzy inference system," *Expert Systems with Applications*, vol. 31, no. 3, 2006, pp. 495-503.
14. Ghosh, Ashish, B. Uma Shankar, and Saroj K. Meher, "A novel approach to neuro-fuzzy classification," *Neural Networks*, vol. 22, no. 1, 2009, pp. 100-109.
15. Iyatomi, Hitoshi, and Masafumi Hagiwara, "Adaptive fuzzy inference neural network," *Pattern Recognition*, vol. 37, no. 10, 2004, pp. 2049-2057.
16. Takagi, Tomohiro, and Michio Sugeno, "Derivation of fuzzy control rules from human operator's control actions," *IFAC Proceedings Volumes*, vol. 16, no. 13, 1983, pp. 55-60.
17. Jang, J-SR, "ANFIS: Adaptive-network-based fuzzy Inference system," *IEEE transactions on systems, man, and cybernetics*, vol. 23, no. 3, pp. 665-685, 1993.
18. Suparta, W and Alhasa, K. M, "Adaptive neuro-fuzzy interference system," *In Modeling of Tropospheric Delays Using ANFIS. Springer, Cham*, 2016, pp. 5-18.
19. Laxmi Sree, B. R. and Vijaya, M. S, "Graph Cut Based Segmentation Method for Tamil Continuous Speech," *In Annual Convention of the Computer Society of India, Springer, Singapore*, December 2016, pp. 257-267.
20. Pushpa, M., and Karpagavalli, S., "Multi-label classification: Problem transformation methods in Tamil phoneme classification," *Procedia Computer Science*, vol.115, 2017, pp.572-579.
21. Madhavaraj, A., & Ramakrishnan, A. G., "Design and development of a large vocabulary, continuous speech recognition system for Tamil," *In Proc. 14th IEEE India Council International Conference (INDICON)*, December 2017.

AUTHORS PROFILE

Laxmi Sree B.R has 13 years of teaching experience and 12 years of research experience.. She has completed M.C.A., M. Phil. She is persuing doctoral programme in PSGR Krishnammal College for Women, Coimbatore. She has published around 20 papers in International conferences and journals. SheHer current research is concerned with Speech Recognition. Her area of interest falls in Pattern recognition, Machine learning, Design and Analysis of Algorithms, Optimization Techniques, Big data and Data Analytics. She has also worked as a principal investigator in UGC funded minor research project. She has presented few invited lectures on Data Structures, NoSQL databases, etc.

Vijaya M.S. has 31 years of teaching experience and 14 years of research experience. She is currently with PSGR Krishnammal College for Women, Coimbatore as an Associate Professor and Head in Department of Computer Science (PG). She has completed her doctoral programme in the area of Natural Language Processing. She is a member of Computer Society of India, Advanced Computing and Communications Society (IISc, Bangalore), International Association of Engineers (Hong Kong), International Association of Computer Science and Information Technology (IACSIT, Singapore). She has served as a member of programme committee in several International conferences. Her areas of interest include Data Mining, Support Vector Machine, Machine learning, Pattern Recognition, Natural Language Processing and Optimization Techniques.