Indian Journal of Science and Technology, Vol 8(35), DOI: 10.17485/ijst/2015/v8i35/80681, December 2015
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

A Hierarchical Approach in Tamil Phoneme Classification using Support Vector Machine

S. Karpagavalli[1*] and E. Chandra[2]

[1]Department of Computer Science, PSGR Krishnammal College for Women, Coimbatore - 641004, Tamil Nadu, India; karpagavalli@psgrkc.com
[2]Department of Computer Science, Bharathiar University, Coimbatore - 641046, Tamil Nadu, India; crcspeech@gmail.com

Abstract

Most of the speech recognition systems are designed based on the sub-word unit phoneme which is the basic sound unit of a language. In the proposed work, a novel hierarchical approach based phoneme classification task has been carried out to reduce time complexity and search space. Hierarchical classification of set of Tamil phonemes has been done in three levels. Phoneme boundaries of the given speech utterance are identified using Spectral Transition Measure (STM) and phonemes are separated. Mel-Frequency Cepstral Coefficients (MFCC) are extracted for each phoneme represented by 9 frames including the contextual frames of corresponding phoneme. In each hierarchical level, different number of models is built using Support Vector Machine (SVM) for classifying each phoneme group/phoneme. It is observed from the results that in hierarchical approach phoneme group recognition rate at level 1 and 2 has greatly improved compared to flat classification model. Complexity of search space is significantly reduced at level 2 and level 3 contrasts to flat phoneme classification model. Hierarchical phoneme classifier can be very well employed in phoneme recognition task which is useful in applications such as spoken term detection, out-of-vocabulary detection, named entity recognition, spoken document retrieval.

Keywords: Hierarchial Classification Mel-Frequency Cepstral Coefficients, Spectral Transition Measure

1. Introduction

Phonemes are the smallest units of intelligible sound and phonetic spelling is the sequence of phonemes that a word comprises. The problem of automatic phoneme classification can be used for robust speech recognition, accent/dialect detection, speech quality scoring, etc. Phoneme classification is inherently complex due to number of possible phonemes of a particular language, variability in speakers, dialects, accents, noise in the environment and segmentation errors. In this proposed work, Phonemes of Tamil, an Indian language are considered.

2. Phonemes in Tamil Language

The phonetic units of a Tamil language can be grouped into different categories such as vowels, plosives, fricatives, etc. According to the place and manner of articulation, phonemes can be organized in hierarchical fashion[1]. This motivates the hierarchical phoneme classification strategy rather than complex flat phoneme classification. The Tamil alphabets with the corresponding International Phonetic Alphabet (IPA) symbols[2] are shown in Table 1.

Tamil phonemes are majorly classified into obstruent and sonorant categories. Obstruent is a consonant sound such as that is formed by obstructing airflow, causing a strong gradient of air pressure in the vocal tract. Sonorant is a speech sound that is produced with continuous, non-turbulent airflow in the vocal tract. The hierarchy of phonemes in Tamil language is shown in Figure 1. Stop consonants also known as plosives, are consonant in which the vocal tract is blocked so that all airflow ceases. Fricatives are consonants produced by forcing air through a narrow channel made by placing two articulators close together. This turbulent air flow is called frication.

* Author for correspondence

Affricates are combination of stops and fricatives, i.e., begins as a stop and release as a fricative with the same place of articulation. All these stops, fricatives and affricates fall in the group of obstruents.

Table 1. Tamil alphabets with IPA symbol

Vowels	அ	ஆ	இ	ஈ	உ	ஊ
	a	a:	i	i:	u, ː	u:
	எ	ஏ	ஐ	ஒ	ஓ	ஔ
	e	e:	aɪ̯	o	o:	aʊ̯
Consonants	க்	ங்	ச்	ஞ்	ட்	ண்
	k, g, h	ŋ	s, tʃ	ɲ	ʈ, ɖ, ɽ	ɳ
	த்	ந்	ப்	ம்	ய்	ர்
	t̪, d̪, ð	n	p,b, β	m	j	ɾ
	ல்	வ்	ழ்	ள்	ற்	ன்
	l	ʋ	ɻ	ɭ	r	N
Grantha	ஜ்	ஷ்	ஸ்	ஹ்	க்ஷ்	ஸ்ரீ
	dʒ	ʂ	s	ɦ	kʂ	ʂri

Nasals are allowing the air to escape freely through the nose but not through the mouth, as it is blocked by the lips or tongue. In terms of acoustics, nasals are sonorants, meaning that they do not significantly restrict the escape of air. In addition to vowels, phonetic categorizations of sounds that are considered sonorant include approximants, taps and trills[3].

3. Literature Review

Literature on similar research experiments are given in this section. In[4], they had performed recognition of Lithuanian phoneme groups using group characteristic features. In first level, they used different algorithms, rules and features for the detection of phoneme groups like plosive consonant, fricative consonant, sonant sound using phonetically labeled LTDIGITS corpora. In the second level, precise phoneme recognition is carried out by comparing the phoneme features with the template values of the relevant phoneme group. Their results indicate that hierarchical classification approach improves the recognition accuracy by 3% and 55% reduction of time taken for classification.

In[5], designed hierarchical phoneme recognition system using TIMIT data. They carried out the task in a number of steps: Segmentation, manner of articulation classification and then place of articulation classification using Time-Delay Neural Networks (TDNN) and achieved improved performance than flat model.

In[6], introduced the phoneme classification problem as a data mining task and proposed a dual-domain (time and frequency) hierarchical classification algorithm. They used a Dynamic Time Warping (DTW) based classifier in the top layers and time-frequency features in the lower layer. They cross-validated their method on phonemes from three online dictionaries and achieved up to 35% improvement in classification compared to existing techniques. In[7], performed Hindi phoneme recognition in two stages: Broad acoustic classification of a frame is followed by fine acoustic classification. A semi-Markov model processed the frame level outputs of a broad acoustic maximum likelihood classifier to provide a sequence of segments with broad acoustic labels. In second stage, segments of each group phoneme decoded by class-dependent neural nets.

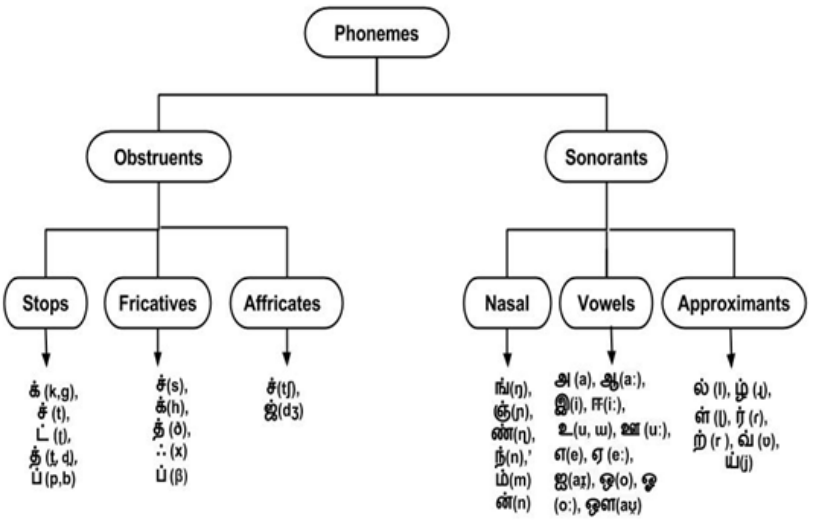

Figure 1. A hierarchy of phonemes in Tamil language.

In[8], designed a phoneme recognition system for Portuguese language using wavelet packet transform with sub-bands selected through the Mel scale. Support Vector Machine had been used in the structure of hierarchical committee machine. They showed a 3.5% improved performance.

4. Proposed Framework

In the proposed work, hierarchical classification of set of Tamil phonemes has been carried in three levels. Phoneme boundaries of the given speech utterance are identified using spectral transition measure and phonemes are segmented. As Support Vector Machine uses only fixed length descriptors for classification, for each phoneme representation, 9 frames are used which includes 4 left context frames, 1 centre frame corresponding to the phoneme and 4 right context frames[9–12]. MFCC features are extracted for each phoneme represented using 9 frames. In Support Vector Machine, for each hierarchical level different number of classifiers is built and the performance of the models are tested for its prediction accuracy at each level. The block diagram of proposed framework is shown in Figure 2.

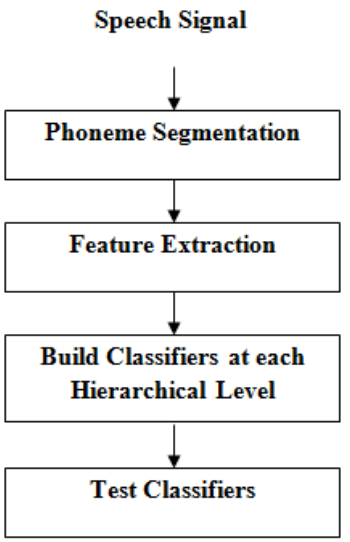

Figure 2. Block diagram of proposed framework.

5. Phoneme Segmentation

Since the problem concentrates on the sub-word unit phoneme, it requires segmentation of speech waveform into phonemes. Speech waveform can be segmented into phoneme manually easily if the dataset is small. For large datasets, it becomes tedious; hence some automatic phoneme segmentation techniques like blind segmentation, spectral transition measure, maximum likelihood segmentation, temporal decomposition, agglomerative segmentation to be adopted. Spectral Transition Measure (STM) is used to approximately identify the boundaries of the phoneme in the given speech waveform. This method is based on maximum spectral rate of change which usually displays peaks at the transition between phonemes. The small/insignificant peaks have been adjusted using different threshold values to fine tune the phoneme boundary identification.

5.1 Spectral Transition Measure

A parameter vector representing spectral features of the signal at time t can be $X(t)=[x_1(t), x_2(t), ...x_p(t)]'$. The spectral gradient or a spectral transition measure is given by the derivative of x(t), x'(t) as $x'(t)=\frac{\partial x(t)}{\partial t}$. Though defined for a continuous case, the derivative x'(t) can be viewed as a vector-valued distortion between infinitesimally small contiguous acoustic vector sequence. The magnitude of x'(t), ||x'(t)||, is a scalar measure and represents the rate at which the spectral feature vector x(t) is changing at time t. This scalar measure (derivative vector magnitude) is expected to be large at the boundaries between successive quasi-stationary speech sounds, corresponding to a transition; these are at the instances of sharp rate of change of the vector trajectory in the p dimensional parameter space representing rapid vocal tract changes.

The acoustic observation is a discrete-time vector sequence, given by be $X(n) = [x_1(n), x_2(n), ...x_p(n)]'$,n = 1,. . .,T, representing a parameter vector X(n) at discrete time instance or frame 'n'. Here, the derivative x'(n) is approximated by differences, the successive difference estimate of x'(n) could be very noisy because of errors due to the parameter estimation process. Hence a low order polynomial fit is used to estimate the trajectory of each vector element. The gradient is obtained by mmse fitting of a straight line to each of the vector elements, within a defined time window. This leads to the slope estimate given by,

$$\Delta x_m(n)=\frac{\sum_{k=-K}^{K} k h_k x_m(n+k)}{\sum_{k=-K}^{K} h_k} \quad (1)$$

Where, h_k is a symmetric window of length (2K + 1). A sufficiently good estimate of the spectral transition can be obtained by using the above gradient estimates. A scalar measure for the spectral variation (or the spectral derivative magnitude ||x'(t)|| is defined as,

$$d\Delta(n) = \sum_{m=1}^{p} (\Delta x(n))^2 \qquad (2)$$

This is a running estimate of gradient vector norm and it tends to exhibit peaks at the boundaries between speech sounds corresponding to changing vocal tract configuration and is hence a measure of its non-stationarity. The problem of finding segment boundaries, between two stationary segments, thus reduces to a peak-picking procedure on dΔ(n),n = 1,...T; these segment boundaries are set to be at the instances of maximum spectral change, i.e., maximum dΔ(n)[13,14].

5.2 Feature Extraction

Mel Frequency Cepstral Coefficients (MFCCs) are the parameterization of choice for many speech recognition applications[15]. They give good discrimination and lend themselves to a number of manipulations. MFCCs are computed from digitized speech signal sampled at 16000Hz. The speech signal is first pre-emphasized using a first order Finite Impulse Response (FIR) filter with pre-emphasis coefficient α =0.97. The pre-emphasized speech signal is subjected to framing and windowing operation with frame duration of 25ms, frame shift of 10ms using hamming window.

The short-time Fourier transform analysis performed after windowing to compute magnitude spectrum. It is followed by filter bank design with triangular filters uniformly spaced on the mel scale between 300 Hz to 3400 Hz as lower and upper frequency limits. The filter bank is applied to the magnitude spectrum values to produce Filter Bank Energies (FBEs) 20 per frame. Log-compressed FBEs are then de-correlated using the Discrete Cosine Transform (DCT) to produce cepstral coefficients. The co-efficients are rescaled to have similar magnitude achieved through liftering with L = 22. The steps involved in MFCC feature extraction are shown in Figure 3.

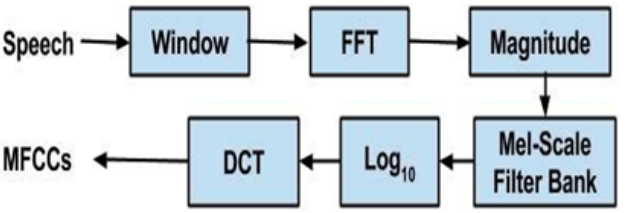

Figure 3. Block diagram of MFCC feature extraction.

6. Support Vector Machine

Support Vector Machines (SVM) is supervised learning model with associated learning algorithms that examine

Table 2. Performance of the phoneme classifier at level I

Classifier	Phoneme Category	SVM Kernel	Correctly Classified Instances	Incorrectly Classified Instances	Predictive Accuracy (in %)
C1 Binary Classifier	Obstruent / Sonorant	Linear	2200	200	91.66
		Polynomial	2255	145	93.95
		RBF	2320	80	96.66

Table 3. (a) Performance of the phoneme classifier at level II (b) Performance of the phoneme classifier at level II

(a)

Classifier	Phoneme Category	SVM Kernel	Correctly Classified Instances	Incorrectly Classified Instances	Predictive Accuracy (in %)
C2 Multi-class Classifier	Stops / Fricatives / Affricates	Linear	908	102	82.54
		Polynomial	938	162	85.27
		RBF	988	112	89.81

(b)

Classifier	Phoneme Category	SVM Kernel	Correctly Classified Instances	Incorrectly Classified Instances	Predictive Accuracy (in %)
C3 Multi-class Classifier	Nasal / Vowel / Approximants	Linear	1087	213	83.61
		Polynomial	1122	178	86.30
		RBF	1168	132	89.84

Table 4. (a) Performance of the phoneme classifier at level III (b) Performance of the phoneme classifier at level III (c) Performance of the phoneme classifier at level III (d) Performance of the phoneme classifier at level III (e) Performance of the phoneme classifier at level III (f) Performance of the phoneme classifier at level III

(a)

Classifier	Phoneme Category - Stop	SVM Kernel	Correctly Classified Instances	Incorrectly Classified Instances	Predictive Accuracy (In %)
C4 Multi-class Classifier	க் (k,g), ச் (t), ட் (ʈ), த் (t̪, d̪), ப் (p,b)	Linear	321	129	71.33
		Polynomial	337	113	74.89
		RBF	355	95	78.80

(b)

Classifier	Phoneme Category - Fricatives	SVM Kernel	Correctly Classified Instances	Incorrectly Classified Instances	Predictive Accuracy (in %)
C5 Multi-class Classifier	ச்(s), க்(h), த் (ð), ஃ (x), ப் (β)	Linear	250	100	71.42
		Polynomial	261	89	74.57
		RBF	268	82	76.57

(c)

Classifier	Phoneme Category - Affricates	SVM Kernel	Correctly Classified Instances	Incorrectly Classified Instances	Predictive Accuracy (in %)
C6 Multi-class Classifier	ச்(tʃ), ஜ்(dʒ)	Linear	218	82	72.67
		Polynomial	221	79	73.66
		RBF	232	68	77.33

(d)

Classifier	Phoneme Category - Nasals	SVM Kernel	Correctly Classified Instances	Incorrectly Classified Instances	Predictive Accuracy (in %)
C7 Multi-class Classifier	ங்(ŋ), ஞ்(ɲ), ண்(ɳ), ந்(n), ம்(m), ன்(n)	Linear	291	109	72.75
		Polynomial	290	110	72.50
		RBF	311	89	77.75

(e)

Classifier	Phoneme Category - Vowels	SVM Kernel	Correctly Classified Instances	Incorrectly Classified Instances	Predictive Accuracy (in %)
C8 Multi-class Classifier	அ (a), ஆ(aː), இ(i), ஈ(iː), உ(u, ɯ), ஊ (uː), எ(e), ஏ (eː), ஐ(aɪ), ஒ(o), ஓ(oː), ஔ(aʊ)	Linear	461	139	76.83
		Polynomial	461	139	76.83
		RBF	472	128	78.67

(f)

Classifier	Phoneme Category - Approximants	SVM Kernel	Correctly Classified Instances	Incorrectly Classified Instances	Predictive Accuracy (in %)
C9 Multi-class Classifier	ல் (l), ழ் (ɻ), ள் (l̪), ர் (ɾ), ற் (r), வ் (ʋ), ய்(j)	Linear	219	81	73
		Polynomial	225	75	75.00
		RBF	235	65	78.33

data and identify patterns. A support vector machine develops a hyper plane or number of hyper planes in a high or unlimited dimensional space, which is used for classification. If hyper plane is achieved the good separation and consists of the largest distance to the nearby training data points of any class. When the classifier has the larger margin then generalization error will be lower. Assigned a set of training examples, each one marked for belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one group or the other, making it very a non-probabilistic binary linear classifier[16].

Instead of creating several binary classifiers, a more natural way is to distinguish all classes in one single optimization processing[17]. For a k-class problem, these methods design a single objective function for training all k-binary SVMs simultaneously and maximize the margins from each class to the remaining ones. Given a labelled training set represented by $\{(x_1,y_1),...,(x_1,y_1)\}$ of cardinality l, where $x_i \in R^d$ and $y_i \in \{1,...,k\}$, the formulation proposed in[18] is given as follows:

$$\min \frac{1}{2}\sum_{m=1}^{k} w_m^T w_m + C\sum_{i=1}^{l}\sum_{t\neq y_i}\xi_{i,t} \qquad (3)$$

$$w_m \in H, b \in R^k, \xi \in R^{lxk}$$

$$subject\ to\ w_{y_i}^T \varphi(x_i) + b_{y_i} \geq w_t^T \varphi(x_i) + b_t + 2 - \xi_{i,t}$$

$$\xi_{i,t} \geq 0,\ i = 1,...,l,\ t \in \{1,...,k\}\backslash y_1$$

The resulting decision function is

$$\text{argmax}_m f_m(x) = \text{argmax}_m (w^T{}_M \varphi(x) + b_m) \qquad (4)$$

7. Experiment and Results

7.1 Dataset

The speech corpus prepared by using 15 native speakers of Tamil, where each speaker uttered each word 5 times. Nearly 100 words of Tamil language covering almost all phonemes present in level 3 of Figure 1 have been used in recordings. In this work, Audacity digital audio editor and high quality microphone was used in recording the data. The recordings carried out in quiet room environment and at sampling rate 16 KHz with 16 bit PCM.

In the first level of hierarchical phoneme classification, 1100 instances of obstruent class and 1300 instances of sonorant class dataset was prepared and used to build a binary classifier C1 in support vector machine, which broadly classifies the given 9 frame phoneme data into obstruent or sonorant class.

In second level, C2 and C3 multiclass classifiers are built to further classify the broadly classified obstruent phoneme as stops/fricatives/affricates or sonorant phoneme as nasals/vowels/approximants respectively.

In third level, C4, C5, C6 multiclass classifiers are built to perform fine classification of phoneme which falls within stops/fricatives/affricates category. Also, C7, C8, C9 multiclass classifiers are built to do precise classification of phoneme which falls within nasals/vowels/approximants category.

The results of the experiment are tabulated in Table 2, 3 (a-b) and 4 (a-f).

8. Conclusion

Phoneme is an indivisible unit of sound in a given language. Tamil language has approximately 45 unique phonemes. The phoneme classification tasks are useful in many speech applications and improve the performance of recognition. In this work, a hierarchical approach in Tamil phoneme classification has been attempted and carried out in three stages. Support Vector Machine has been used to build the different number of models in each stage. This hierarchical approach significantly improves performance of phoneme recognition as well as the time taken for classification also drastically reduced. In future, to increase the recognition rate in third level, template based approach can be adopted or hybrid models which uses templates as well as specific features that uniquely differentiates each phoneme category can be used.

9. References

1. Ilakkuvanar S. Tholkappiyam (in English). India: Kural Neri Publishing House; 1963.
2. Keane E. Tamil Journal of the International Phonetic Association. 2004; 34(1):111–6.
3. Karunakaran K, Jeya V. moZhiyiyal (in Tamil). India: Kavitha Pathippakam; 1997.
4. Driaunys KK, Rudzionis VV, Zvinys PP. Implementation of hierarchical phoneme classification approach on LT-DIGITS data. Information Technology and Control. 2009; 38(4):303–10.
5. Grayden DB, Scordilis MS. Recognition of obstruent phonemes in speaker independent fluent speech using a hierarchical approach. Proceedings of 3rd European conference on speech communication and technology, Eurospeech'93; 1993. p. 855–8.
6. Hamooni H, Mueen A. Dual-domain hierarchical classifi-

cation of phonetic time series. Proceedings of 2014 IEEE International conference on Data Mining; Shen. 2014. p. 160–9.
7. Samudravijaya K, Ahuja R, Bondale N, Jose T, Krishnan S, Poddar P, Rao PVS, Raveedran R, Ahuja R, Bondale N, Jose T, Krishnan S, Poddar P, Rao PVS, Raveedran R. A feature-based hierarchical speech recognition system for Hindi. Academy Proceedings in Engineering Sciences. 1998; 23(4):313–40.
8. Bresolin AA, Neto ADD, Alsina PJ. A new hierarchical decision structure using wavelet packet and SVM for Brazilian phonemes recognition. Proceedings of the 13th International Conference on Neural Information Processing; 2006. p. 159–66.
9. Renals SS, Rohwer RR. Phoneme classification experiments using radial basis functions. International Joint Conference on Neural Networks; Washington DC, USA. 1989. p. 461–7.
10. Khoo L, Cvetkovic Z, Sollich P. Robustness of phoneme classification in different representation spaces. 14th European Signal Processing Conference; Florence. 2006. p. 1–5.
11. Pinto J, Yegnanarayana B, Hermansky H, Magimai-Doss M. Exploiting contextual information for improved phoneme recognition. IEEE International Conference on Acoustics, Speech and Signal Processing; Las Vegas, USA. 2008. p. 4449–52.
12. Karpagavalli S, Chandra E. Tamil phoneme classification using contextual features and discriminative models. 2015 IEEE International Conference on Communications and Signal Processing; Melmaruvathur. 2015 Apr 2-4. p. 564–8.
13. Sai Jayram AKV, Ramasubramanian V, Sreenivas TV. Robust parameters for automatic segmentation of speech. IEEE International Conference on Acoustics, Speech, and Signal Processing; Orlando, FL, USA. 2002 May 13-17. p. 513–6.
14. Dusan S, Rabiner L. On the relation between maximum spectral transition positions and phone boundaries. Proceedings of 9th International Conference on Spoken Language Processing; 2006. p. 645-1-4.
15. Rabiner R, Juang B-H. Fundamentals of speech recognition. New Jersey: Prentice-Hall International; 1993.
16. Soman KP, Loganathan R, Ajay V. Machine learning with SVM and other kernel methods. India: PHI; 2009.
17. Vapnik V. Statistical learning theory. New York: Wiley-Interscience; 1998.
18. Weston J, Watkins C. Multi-class support vector machines. In: Verleysen M, editor. Proceedings of European Symposium on Artificial Neural Networks; Brussels, Belgium. 1999. p. 1–9.