Morpheme Based Language Model for Tamil Speech Recognition System

Selvarajan Saraswathi and Thekkumpurath Geetha
Department of Computer Science and Engineering, Anna University, India

Abstract: *This paper describes the design of a morpheme based language model for Tamil language. It aims to alleviate the main problems encountered in processing the Tamil language, like enormous vocabulary growth caused by large number of different forms derived for one word. The size of the vocabulary is reduced by decomposing the words into stems and endings and storing these sub word units (morphemes) for training the language model The modified morpheme based language model was applied to avoid the ambiguities in the recognized Tamil words. The perplexity, Out Of Vocabulary (OOV) rate and Word Error Rate (WER) parameters were obtained to check the efficiency of the model for Tamil speech recognition system. The results were compared with the traditional word based statistical bigram and trigram language models. From the results, it was analyzed that the modified morpheme based trigram model with Katz back off smoothing effect improved the performance of the Tamil speech recognition system when compared to the word based N-Gram language models.*

Keywords: *Language model, morphemes, perplexity, out of vocabulary rate, word error rate.*

Received November 29, 2005; accepted April 18, 2006

1. Introduction

In this paper, a new approach to language modeling which is suitable for highly inflectional languages like Tamil and other south Indian Dravidian languages is proposed. When we try to build large vocabulary speech recognition system for these languages we encounter a major problem of excessive vocabulary growth caused by great number of different word forms derived from a single root word. An inflectional change of the word predominantly affects word endings, whereas stem remains constant. However, the number of different endings is relatively very small, whose combinations with the stems leads to large number of derived word forms. In this paper, the words are decomposed into stems and endings and these unit's are treated separately as independent units. By segmenting words into morphemes we can improve the performance of natural language systems including machine translation [5], information retrieval [8] and speech recognition [15].

The morphological rules are used to produce subword units in highly inflected languages [6, 10, 11]. The rules are based on language dependent prior assumptions about stems and suffixes. Morphological analysis of highly inflected languages can be performed using two methods [11]. In one of the methods, the coverage of subword units is maximized given the lexicon size. In the other, the most common words are taken as a basis and the rest of the words are generated using a combination of the most common words and sub-words units. This work is based on the second method of grouping the common sub-words and then generating the rest of the words from the combination of these words with the suffixes or word endings.

This paper describes the characteristics of morpheme based language models. It also provides the design of a modified morpheme based language model for Tamil speech recognition system and compares the performance of the proposed model with word based statistical N-Gram language models.

1.1. Features of Tamil Language

There are two important differences between Tamil and English that are of relevance to statistical language modelling and that are shared in varying degrees by many other languages. They are word formation and word ordering. Tamil words typically have more morphological patterns than English words. For example, a Tamil word will often contain the following easily identifiable, constituent parts: A stem, which can be thought of as responsible for the nuclear meaning of the verb, attached to which may be zero or more derivational prefix(es) and zero or one suffix(es), which together form a word. The stem often acquires an entirely new lexical meaning with the presence of these affixes.

Of most relevance to language modelling, however, is the inflection (inflectional suffix), which is appended to the stem and which determines the grammatical case, gender (masculine, feminine, or neuter), number, etc. of the word. The presence of the inflection results in many different word forms for a word in Tamil compared to English [3]. The direct consequence of this is the coverage of more words in Tamil vocabulary than that of the same sized English vocabulary. English compensates for having less grammatical information encoded within the words themselves, by imposing strict constraints on the relative order of words in a sentence. In the sentence, “the boy kicks the ball”, it is clear who is doing what to whom from the order in which the words are written. In Tamil, on the other hand, the subject and object of the sentence can only be determined by the inflection associated with the words and by agreement with the verb, not from the order of the words themselves. In fact, the above subject sentence translated into Tamil could be expressed by six different permutations for the three words “boy”, “kicks” and “ball” without loss of meaning, just by changing only the endings of the words. Clearly, this phenomenon has the potential for seriously weakening the predictive power of some statistical language models, however, in reality some word orderings are preferred stylistically to others. In particular, a different emphasis is placed on a word depending on its position in the sentence, so the permutations of a sequence of words will actually occur with different frequencies and this led to the design a language model suitable for Tamil speech recognition system.

1.2. Language Models in Speech Recognition

In general, the output of the recognition system produces a sequence of phonemes. During recognition, the sequence of symbols generated by the acoustic component is compared with the set of words present in the lexicon to produce optimal sequence of words that compose the system’s final output. Rules are introduced during this stage to describe the linguistic restrictions present in the language and to allow reduction of possible invalid phoneme sequences. This is accomplished through the use of language models in the system.

A language model comprises two main components: The vocabulary which is a set of words that are recognized by the system and the grammar which is a set of rules that regulate the way the words in the vocabulary are arranged into groups to form sentences. The grammars are made of formal linguistic rules. The linguistic model introduces strong restrictions in allowable sequence of words but becomes computational demanding when incorporated in a speech recognition system. They also have the problem of not allowing the appearance of grammatically incorrect sentences that are often present in spontaneous speech. This makes the stochastic models based on probabilities for sequences of words more attractive for use in speech recognition system due to their robustness and simplicity. The word based statistical N-Gram language models and the morpheme based language models were used in this work to reduce the error rate in the Tamil speech recognition system.

In section 2, Tamil text and speech corpora used for recognition are discussed. In Sections 3 and 4, the use of N-Gram and morpheme based language models for Tamil speech recognition system is explained. In section 5, the design of the modified morpheme based language model for Tamil language is discussed. In the final section, the results are analyzed.

2. Tamil Speech and Text Corpora

For performing language modelling, the text corpus was collected from two different domains. The language modelling was done on the text corpus. The performance of the modelling was tested on the speech corpus collected with respect to these domains.

2.1. Text Corpus

For language modeling purpose, text from newspapers and magazines were collected. The newspaper corpus approximately consists of 5, 00,000 words (5M) and the current political issues corpus, collected from magazines had approximately 2, 00,000 words (2.2M). Text data has to go through several preprocessing stages in order to obtain clear and unambiguous data before statistical and morphological analysis. The punctuation marks that are not usually pronounced in spoken communication are discarded from the training data set.

2.2. Speech Corpus

Read articles on political issues and from newspaper were collected for processing. The news corpus with 200 sentences, containing 2000 words was recorded from 10 persons, 5 males and females. The recording done for 30 minutes, for each speaker yielded five hours of speech information and the political corpus with 100 sentences, containing 900 words was recorded from 10 persons, 5 males and females. The recording done for 15 minutes, for each speaker yielded 2 hours 30 minutes of speech information. Portions of the speech corpus containing long silence or other non-speech material are marked and they were not considered for recognition.

3. Statistical N-Gram Language Model

N-Gram language models are traditionally used in large vocabulary speech recognition systems to provide the recognizer with an a-priori likelihood P (W) of a given word sequence W. The N-Gram language model is usually derived from large training texts that share the same language characteristics as expected input.

The probability of occurrence of a word sequence W is calculated as:

$$
\begin{aligned}
P(W) &= P(w_1, w_2, \cdots w_n) \\
&= P(w_1)P(w_2 / w_1)P(w_3 / w_1 w_2) \cdots P(w_n / w_1 w_2 \cdots w_n) \\
&= \prod_{i=1}^{n} P(w_i / w_{1,} w_2, \cdots w_{i-1})
\end{aligned} \tag{1}
$$

Where W_1, W_2, …, W_n corresponds to sequence of words that form a sentence. The probability of occurrence of a word W_n based on the probability of occurrence of the previous sequence of n - 1 words preceding it forms the design of the N-Gram language model [9]. The N-Gram model uses the previous (*n - 1*) words as the only information source to generate the model parameters. The N-Grams are easy to implement, easy to interface with and good predictors of short-term dependencies [12]. The N-Gram assigns zero probability to words not present in the corpus. There are some perfect N-Grams that are not available in the training corpus. Smoothing is a way of assigning a small but non–zero probability to these zero probability N-Grams. Katz back-off smoothing technique is usually applied for N-Gram approach as it provides improved recognition rates when compared to other smoothing techniques [16]. The back-off models provide an efficient method for increasing coverage and hence overall performance of the system. The bigram language model using Katz back off smoothing technique is calculated as:

$$
P(w_i / w_{i-1}) = \begin{cases} \dfrac{count(w_{i-1} w_i)}{count(w_{i-1})} & if\ count(w_{i-1} w_i) > 0 \\ \alpha\, P(w_i) & otherwise \end{cases} \tag{2}
$$

Where α takes a value in the range 0 - 1.

The value of *α* was set to 0.2 in Bigram language model. The Trigram language model using Katz back off smoothing technique is calculated as:

$$
P(w_i / w_{i-2} w_{i-1}) = \begin{cases} \dfrac{count(w_{i-2} w_{i-1} w_i)}{count(w_{i-2} w_{i-1})} & if\ count(w_{i-2} w_{i-1} w_i) > 0 \\ \alpha_1 \dfrac{count(w_{i-1} w_i)}{count(w_{i-1})} & if\ count(w_{i-1} w_i) > 0 \\ \alpha_2\, P(w_i) & otherwise \end{cases} \tag{3}
$$

Where *α_1 and α_2 takes a value in the range 0-1.*

The number of distinct Unigrams, Bigrams and Trigrams in the training corpus is shown in Table 1. Both Bigram and Trigram models with back-off smoothing effects were applied over the given corpus and the OOV rate, perplexity and WER results as shown in Table 2, were analyzed for values of $\alpha_1 = 0.8$ and $\alpha_2 = 0.2$.

Table 1. Number of N-Grams in the training set.

Corpus	Size	Unigrams	Bigrams	Trigrams
News	5M	1,49,156	1,99,236	2,36,569
Politics	2.2M	53,521	95,933	1,28,630

Table 2. Results of Bigram and Trigram models with Katz backoff smoothing technique.

Corpus	OOV	Bigram		Trigram	
		Perplexity	WER	Perplexity	WER
News	4.5	147.02	14.1	126.98	13.8
Politics	9.2	234.73	28.6	226.04	25.04

The number of Unigrams, Bigrams, and Trigrams were high for both the corpora. They can be further reduced and the performance of the recognition system can be improved by the use of morpheme based language models in highly inflected languages like Tamil. The use of morphemes and the design of the modified morpheme based language model for Tamil language are discussed in the next section.

4. Morpheme Based Language Model

One of the main problems with speech recognition systems is that the words spoken during speech recognition task do not exist within the system's vocabulary, often referred to as Out Of Vocabulary (OOV) words. The solution for this problem is to increase the size of the training vocabulary to reduce the OOV rate. But it is very difficult task in highly inflected languages like Tamil to reduce the OOV rate by increasing the vocabulary size because of the great vocabulary expansion caused by large number of different words derived from basic stems. Many words have different forms for singular and plural and may also change with the gender. To get an idea on the difference between English and highly inflectional languages like Tamil, the number of different forms of the verb "to sing" in Tamil and English in the simple present form is shown in Table 3.

Table 3. Comparison of the forms of the verb "to sing" in simple present between English and Tamil Language.

English	Sing Sings
Tamil	ðf´ ðf´è¤øf÷¢ ðf´è¤øfù¢ ðf´è¤øfó¢è÷¢ ðf´è¤ø¶ ðf´è¤«øù¢ ðf®è¢ªèfí¢®¼è¢è¤«øù¢ ðf®è¢ªèfí¢®¼è¢è¤øfó¢è÷¢

In English, for the two different forms (sing, sings), in Tamil, there are eight forms, based on the gender and the number of persons involved in action. The root verb being "ðf´" with different types of endings "è¤øf÷¢", "è¤øfù¢", "è¤øfó¢è÷¢", "è¤ø¶", "¢ªèfí¢®¼è¢è¤«øù¢¢" and "ªèfí¢®¼è¢è¤øfó¢¢è÷¢". The size of the inflectional languages can be reduced by storing only the root word "ðf´" as the stem word and all the possible endings of the stem "ðf´" as the ending words.

Table 4. Example of decomposition into stem and ending.

Words	Decomposition	Translation
ï¤èö¢ê¢ê¤è÷¢	ï¤èö¢ê¢ê¤ # è÷¢ $	Programs
ÞÁî¤ò¤ô¢	ÞÁî¤ # ò¢ $ Þô¢ $	In the end
ñ£ï¤ôî¢î¤ô¢	ñ£ï¤ôñ¢ # Üî¢¶ $ Þô¢ $	In the state
è£ôî¢î¤ô¢	è£ôñ¢ # Üî¢¶ $ Þô¢ $	In time
Þòé¢°è¤ù¢øù	Þòé¢° # è¤ù¢Á $ Üù $	In functional status

Decomposition of the Tamil text corpus into stems and endings is done using an existing Tamil morphological analyser [2]. Stems and endings are marked with different marks (stems with # sign and endings with $ sign) in order to allow them to be co-joined back after recognition. Examples of decomposition are given in Table 4. The size of the morpheme vocabulary generated for the Tamil corpus is shown in Table 5. Usage of this model leads to significant reduction in the size of the corpus.

Table 5. Size of morpheme vocabulary.

Corpus	No. of Distinct Stem	No. of Distinct Endings	Reduction in Vocabulary Size
News	93,795	1515	63.9
Politics	30,219	1273	58.84

In the general morpheme based language model, morphemes are treated as if they are independent words [7, 13]. No distinction is made between stems and endings. The major flaws in morpheme based language model was detected while assigning probabilities to the Morphemes. Considering the following decomposition:

Word w_{i-1} decomposed as stem s_{i-1} and ending e_{i-1}
Word w_i decomposed as stem s_i and ending e_i

Prediction of the $i - 1^{th}$ word ending e_{i-1}, is based on the knowledge of the corresponding stem s_{i-1} *only*, i. e.,

$$P (e_{i-1} / s_{i-1})$$

Such type of dependency of the ending for a stem has a strong dependency between the morphs, because particular stem can be followed by relatively small set of endings only. Prediction of the i^{th} word stem s_i, is based on the knowledge of the preceding ending e_{i-1}, only i. e.,

$$P (s_i / e_{i-1})$$

In this case, the dependency is very weak, because the ending bears information about grammatical properties of the word (such as case of the noun, person of the verb, etc.), not information about the word itself. So the probability estimates for the general morpheme based language model does not provide strong dependency among the morpheme sequence generated from the corpus and it will not lead to good performance results for the inflectional languages. Therefore some modifications are done on the morpheme based language model.

5. Modified Morpheme Based Language Model

As argued in the previous section, prediction of the stem s_i should not only be based on the knowledge of the preceding ending e_{i-1}, but also dependent on the previous stem s_{i-1}, i. e., the language model should also consider the probability $P (s_i / s_{i-1})$ to stem s_i. Since the stem gives the major part of the information about the word, quality of such dependency should be comparable to word bigram. Prediction of the ending is more complicated. Ending e_i should depend on the corresponding stem s_i. In addition, the Tamil language makes extensive use of agreement, for example a noun and its adjectival or pronominal attribute must agree in gender, number and case. The morphological categories often affect word-ending e_i. So the ending e_i of the word w_i should also be based on ending e_{i-1} of the preceding word w_{i-1}.

Consider the following decomposition of word w_i and w_{i-1}:

Word w_{i-1} decomposed as stem s_{i-1} and ending e_{i-1}
Word w_i decomposed as stem s_i and ending e_i

According to the modified morpheme based language model, the prediction of stem s_i depends on s_{i-1} and also e_{i-1}:

$$P (s_i) \text{ depends on } P (s_i / s_{i-1}) \text{ and } P (s_i / e_{i-1})$$

and prediction of the ending e_i depends on the previous ending e_{i-1} and the stem s_i:

$$P (ei) \text{ depends on } P (ei / ei - 1) \text{ and } P (ei / si)$$

Since there is a strong dependency between the stem and its endings, all possible endings of the stems present in the training corpus is found using the existing Tamil morphological generator [1]. The generator generates all the possible endings for the given stem word. All stem - ending combinations are used in evaluating the probability of occurrence of the morphs in the modified morpheme based language model.

The bigram probability estimation for the modified morpheme based language is calculated for stems as follows:

$$P(S) = \alpha.P(s_i / s_{i-1}) + (1-\alpha).P(s_i / e_{i-1}) \tag{4}$$

And the bigram probability estimation for the endings is calculated as:

$$P(E) = \xi.P(e_i / s_i) + (1-\xi).P(e_i / e_{i-1}) \qquad (5)$$

Where α and ξ are parameters in the range *0 to 1*. The knowledge of the preceding ending gives less information on the occurrence of a stem and also the occurrence of the preceding ending gives less information on the occurrence of the next word ending. So, the value of α and ξ were set to 0.9, for which improved perplexity values were obtained. The bigram probability of occurrence of a word based on the stem-end combination was calculated by:

$$P(w_i / w_{i-1}) = \begin{cases} P(S)P(E) & \textit{if } cnt(s_i e_i) \textit{ and } cnt(s_{i-1}e_{i-1}) > 0 \\ P(e_i / s_i) & \textit{if } cnt(s_i e_i) > 0 \textit{ and } cnt(s_{i-1}e_{i-1}) = 0 \\ P(s_i) & \textit{otherwise} \end{cases} \qquad (6)$$

The probability of occurrence of a word is based on the combined probability of occurrence of its stem and endings. If it is not possible to identify the word, based on the stem-ending combinations of the words W_i and W_{i-1}, the number of occurrence of the word (W_i) with the stem (s_i) and ending (e_i) pair is estimated. If no words exist with that stem-ending combination then the number of occurrence of the stem (s_i) of the word (W_i) is estimated. The bigram probability of the occurrence of the word W_i, after smoothing is represented as:

$$P(w_i / w_{i-1}) = \varphi 1 P(S)P(E) + \varphi 2 P(e_i / s_i) + \varphi 3 P(s_i) \qquad (7)$$

Where $\phi 1 + \phi 2 + \phi 3 = 1$.

For values $\phi 1 = 0.5$, $\phi 2 = 0.3$ and $\phi 3 = 0.2$ improved perplexity and WER were obtained. The results obtained for the modified morpheme based language model is shown in Table 6. The modified morpheme based Trigram language model showed an improvement in the perplexity and WER values, than the statistical word based N-Gram language models.

Table 6. Results of modified morpheme based Bigram and Trigram models using Katz backoff smoothing technique.

Corpus	OOV	Modified Morpheme Based Bigram		Modified Morpheme Based Trigram	
		Perplexity	WER	Perplexity	WER
News	2.6	125.36	13.5	113.2	12.9
Politics	7.8	186.87	26.8	146.08	23.9

6. Performance Analysis

The speech signals are first segmented at phonetic level based on the acoustic characteristics [4, 14]. The segmented phonemes were then mapped to their corresponding grapheme sequence. The graphemes were combined to form word sequences. Ambiguities in the word sequences were detected and corrected using the N-Gram and modified morpheme based language models. For both the news and the politics corpora the modified morpheme based Trigram model with Katz backoff smoothing technique gave improved perplexity and word error rate when compared to the N-Gram language models. Comparison of the performance of the different language models with respect to their perplexity and WER is shown in Figure 1 and Figure 2. The variation in perplexity with respect to WER is shown in Figure 3. A decrease in perplexity value results in avoiding the ambiguities in the recognition process and leads to a reduction in WER.

There is a significant reduction in OOV rate, perplexity and WER for the news corpus when compared with the politics corpus. The main reason for this was due to the corpora used for training. The news corpus was trained for more data than the politics corpus. The size of the news text corpus used for training purpose was 40% higher than the size of the politics text corpus used for training. So the OOV rate and the perplexity values are less for it when compared to the politics corpus.

7. Conclusion

In this paper, a new language model based on morphs is designed. The performance of the modified morpheme based language model was compared with the general word based N-Gram models. The performance of the language models were evaluated on two different speech corpora's - news and politics. The modified morpheme based Trigram language model with Katz backoff smoothing technique gave improved perplexity and WER for the two Tamil corpora. The results have shown that the proposed modified morpheme based language model is best suited for Tamil language. The future scope of this work is to test this technique on large test sets from various domains to prove their robustness.

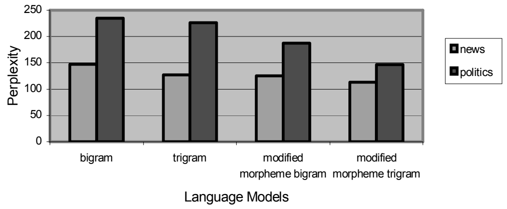

Figure 1. Comparison of perplexity values in two different corpora for different language models.

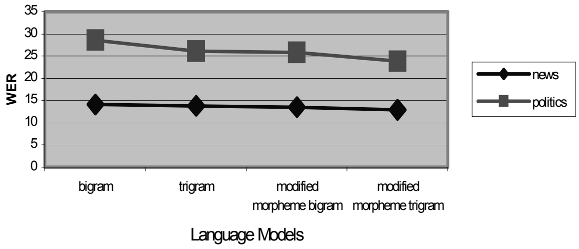

Figure 2. Comparison of WER values in two different corpora for different language models.

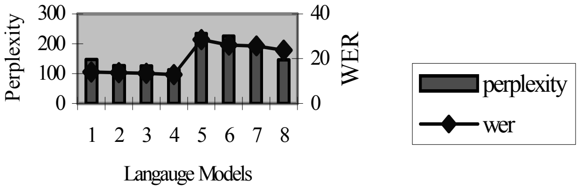

Figure 3. Comparison of perplexity and WER values obtained using different language models.

References

[1] Anandan P., Geetha T. V., and Parathasarathy R., "Morphological Generator for Tamil," *in Proceedings of Tamil Inayam Conference*, Malaysia, pp. 46-50, 2001.

[2] Anandan P., Saravanan K., Parthasarathi R., and Geetha T. V., "Morphological Analyzer for Tamil," *in Proceedings of ICON'2002*, RCILTS-Tamil Anna University, Chennai, 2002.

[3] Arden R. A. H. and Clayton A. C., "A Progressive Grammar of the Tamil Language," Christian Literature Society, Madras, 1969.

[4] Aversions G., Esposito A., Esposito A., and Marinaro M., "A New Text Independent Method for Phoneme Segmentation," *in Proceedings of the IEEE International Workshop on Circuits and Systems*, Dayton, Ohio, vol. 2, pp. 516-519, 2001.

[5] Brown P., Pietra S., Della Pietra V., and Mercer R., "The Mathematics of Statistical Machine Translation: Parameter Estimation," *Computational Linguistics,* vol. 19, no. 2, pp. 263-311, 1993.

[6] Byrne W., Hacic J., Iircing P., Jelinek F., Khudanpur S., Krbec P., and Psutka J., "On Large Vocabulary Continous Speech Recognition of Highly Inflectional Language-Czech," *in Proceedings of Eurospeech'2001,* Aalborg, Denmark, pp. 487-489, 2001.

[7] Creutz M. and Lagus K., "Unsupervised Discovery of Morphemes," *in Proceedings of the Workshop on Morphological and phonological Learning of ACL'2002*, Philadelphia, PA, pp. 21-30, 2002.

[8] Franz M. and McCarley S., "Arabic Information Retrieval," *in Proceedings of TREC'2002*, Gaithersburg, Maryland, pp. 402-405, 2002.

[9] Huang X., Acero A., and Hon H. W., *Spoken Langauge Processing*, Prentice Hall, 2001.

[10] Huckvale M. and Fang A., "Using Phonologically Constrained Morphological Analysis in Continuous Speech Recognition," *Computer Speech and Language*, vol. 16, no. 2, pp. 165-181, 2002.

[11] Kneissler J. and Klakow D., "Speech Recognition for Huge Vocabularies by Using Optimized Sub-Word Units," *in Proceedings of Eurospeech'2001*, Denmark, pp. 69-72, 2001.

[12] Laferty S., and Suhm B., "Cluster Expansion and Iterative Scaling of Maximum Entropy Language Models," *in Proceedings of the 15th International Workshop on Maximum Entropy and Bayesian Methods*, Santa Fe, New Mexico, 1995.

[13] Saraswathi S. and Geetha T. V., "Building Language Models for Tamil Speech Recognition System," *in Proceedings of AACC'2004*, Nepal, pp. 161-168, 2004.

[14] Saraswathi S., Geetha T. V., and Saravanan K., "Integrating Language Independent Segmentation and Language Dependent Phoneme Based Modeling for Tamil Speech Recognition," *Asian Journal of Information Technology*, vol. 5 no. 1, pp. 38-43, 2006.

[15] Siivola V., Hirsimaki T., Creutz M., and Kurimo M., "Unlimited Vocabulary Speech Recognition Based on Morphs Discovered in an Unsupervised Manner," *in Proceedings of Eurospeech'2003*, Geneva, Switzerland, pp. 2293-2296, 2003.

[16] Wu J. and Zheng F., "On Enhancing Katz Smoothing Based on Back-off Language Model," *in Proceedings of the ICSLP'2000*, Beijing, China, vol. 1, pp. 198-201, 2000.

Selvarajan Saraswathi is an assistant professor, in the Department of Computer Science and Engineering, Pondicherry Engineering College, Pondicherry, India. Currently, she is doing her PhD in Anna University, in the area of speech recognition for Tamil language. Her areas of interest include speech processing, artificial intelligence and expert systems.

Thekkumpurath Geetha is a professor in the Department of Computer Science and Engineering, Anna University, India. She has twenty years of teaching experience and has supervised six PhD students so far. She is interested in the area of Tamil computing and has done projects for Ministry of Information Technology, Government of India that includes development of Tamil corpora, Tamil office suite, speech engine, Tamil search engine and parser for Tamil. Currently, she supervises seven PhD students in the areas of speech processing, information extraction, visualization and game theory. Her research interests include artificial intelligence, speech processing, intelligent systems, and compiler design.