Spoken Language Identification for Indian Languages Using Split and Merge EM Algorithm

Naresh Manwani, Suman K. Mitra, and M.V. Joshi

Dhirubhai Ambani Institute of Information and Communication Technology,
Gandhinagar, India

Abstract. Performance of Language Identification (LID) System using Gaussian Mixture Models (GMM) is limited by the convergence of Expectation Maximization (EM) algorithm to local maxima. In this paper an LID system is described using Gaussian Mixture Models for the extracted features which are then trained using *Split and Merge Expectation Maximization Algorithm* that improves the global convergence of EM algorithm. It improves the learning of mixture models which in turn gives better LID performance. A maximum likelihood classifier is used for classification or identifying a language. The superiority of the proposed method is tested for four languages

1 Introduction

Language Identification (LID), as the name suggests is an issue of identifying the language of any utterance irrespective of its length (duration of speech), context (topic and emotions) and speaker (gender, age and demographic region). "Humans have the best capability to identify the language" [1]. Due to the increasing demand of global communications, it is required to break the boundaries of languages. This gives new challenges to machine translation system of languages and speech recognition system also. For that the first step is identifying the language of the speech. Once a particular language has been identified, a translation or a recognition system can be trained to solve the problem based on the identified language.

LID based on language independent phoneme recognition followed by language modeling (PRLM) [2] needs phoneme recognizer. LID based language dependent parallel phoneme recognition (PPR) [2] requires labeled speech. It needs language dependent phoneme recognizer for each language. Both PRLM and PPR perform very well but are computationally very expensive. Alternate methods which do not require labeled speech have also been proposed but their reliability depends on the speech quality and the parameterization technique.

Parallel syllable like unit recognizers [3] can also be used in place of parallel phoneme recognizer for LID. This approach does not require annotated corpora. But its performance depends on how efficiently speech is segmented into syllables like sounds. Recently Auto Associative Neural Network (AANN) [4] are also used for LID. Which does not require transcribed database, butuses heuristics for modelling. Gaussian Mixture Models (GMM) are also used for LID [2]. Although

A. Ghosh, R.K. De, and S.K. Pal (Eds.): PReMI 2007, LNCS 4815, pp. 463–468, 2007.

performance of this approach is comparable to other approaches, it still suffers from the problem of its convergence to local maxima.

Feature extraction methods play important role in language discrimination. Mel Frequency Cepstral Coefficients (MFCC), Perceptual Linear Predictive (PLP) coefficients, Linear Prediction Coefficients (LPC) etc are some of the most commonly used feature extraction methods in speech applications. Recently new feature extraction techniques such as Modified Group Delay Feature (MGDF)[5], Time Frequency Principal Component (TFPC) [6] are explored.

In this paper, we first extract the MFCC and their delta as well as delta-delta coefficients as the features for the speech utterences. These features are then modelled as GMM and a split and merge EM(SMEM) algorithm is used to obtain the model parameters. The use of SMEM overcomes the difficulty of local maxima dur to EM. We show that the accuracy of the system can be improved by using split and merge EM algorithm.

The rest of the paper is organized as follows: section 2 discusses in brief about the GMM, their learning using EM algorithm and its limitations. In section 3 the split and merge is described which is used to overcome the limitation of EM algorithm. Section 4 shows the experiments and performance results of LID system using SMEM.

2 Gaussian Mixture Models and Expectation Maximization Algorithm

Gaussian mixture models (GMMs) play a very important role in pattern recognition. GMMs are used to approximate the distribution of the data as weighted sum of the multivariate Gaussian probability density function (pdf).

Efficient computation of the maximum likelihood parameter estimates of the GMM can be done with the EM(expectation maximization) algorithm. It optimizes the likelihood that the given data points (feature points as used in this study) are generated by a mixture of Gaussian probability density function [7].

In EM algorithm two steps are repeated iteratively. The first step also called E-step is used to calculate the expected data log-likelihood function. In the second step called M-step estimates of new parameter are obtained by maximizing the log-likelihood function. Finally, these two steps give estimated parameters.

1. EM algorithm breaks down when any Gaussian component has its covariance matrix singular. It happens when clusters contain insufficient observations or too many components are used to fit the data set where there are actually fewer clusters[9].
2. Another limitation of EM algorithm is it does not give the global maximum of the log-likelihood of the data, instated it gives us the local maxima.

3 Split and Merge Expectation Maximization Algorithm

SMEM algorithm was basically proposed by Ueda et al.[8]. It overcomes the problem of local maxima in parameter estimation of mixture models using EM algorithm. The main idea behind SMEM algorithm is that after usual convergence of EM algorithm split and merge operations are performed to update the parameters of some mixture components. Then again EM is performed. This process is repeated iteratively until log-likelihood is increased. The number of components are kept constant. This process improves the global convergence of the EM algorithm. This make GMMs to learn the languages better and the result is better LID performance. Split and merge criterion are described as below.

3.1 Split Criterion

For splitting, a local Kullback divergence can be defined as [8]:

$$J_{split}(m;\Theta) = \int f_m(x;\Theta) log \frac{f_m(x;\Theta)}{p_m(x;\theta_m)} dx, \qquad (1)$$

which is a splitting measure for the mth component of mixture model, $\forall\ \theta$ is the model parameter vector. The above equation actually represents the distance between two distributions: the local data density $f_m(x)$ around the mth model and the density of the mth model specified by the current parameter estimate Θ [8]. The local data density is written as:

$$f_m(x;\Theta) = \frac{\sum_{n=1}^{N} \delta(x-x_n) p(m|x_n;\Theta)}{\sum_{n=1}^{N} p(m|x_n;\Theta)}. \qquad (2)$$

The expression given in Eqn. (2) is a modified empirical distribution weighted by the posterior probability so that the data around the mth model is focused on. When the weights are equal, *i.e.*, $p(m|x_n;\Theta) = 1/M$, then $f_m(x;\Theta)$ becomes $p_m(x;\Theta)$ where:

$$p_m(x;\Theta) = \frac{1}{N} \sum_{n=1}^{N} \delta(x-x_n). \qquad (3)$$

The splitting measure $J_{split}(m;\Theta)$ is calculated for all components in the mixture model and the component corresponding to the maximum value of $J_{split}(m;\Theta)$ has the worst estimate of the local density and this is the best candidate for split.

3.2 Merge Criterion

If there are two mixture components such that the posterior probabilities of several data points belonging to these two components are same, then the two components should get merged. To calculate a suitable measure of this, a merge criterion is defined as follows:

$$J_{merge}(i,j;\Theta) = \frac{p_i(\Theta)^t p_j(\Theta)}{||p_i(\Theta)||\ ||p_j(\Theta)||}, \qquad (4)$$

where $p_i(\Theta) = (p(i|x_1;\Theta), p(i|x_2;\Theta), ..., p(i|x_N;\Theta))^t \in \mathcal{R}^N$ is an N-dimensional vector consisting of the posterior probabilities for data points to belong component $\imath$. t denotes the transpose operation and $||.||$ denotes the Euclidean vector norm. Two components $\imath$ and $\jmath$ with large value of $J_{merge}(i,j;\Theta)$ are supposed to be good candidates for merge.

To get the parameters of the components after split and merge operation a method proposed by Zhang et al. [11] is used.

4 Experimental Results

Testing of thealgorithm has been done on four language viz. English, HIndi, Gujarati and telegu. For English language IViE corpus is used. The statistics of speech samples that are used for training and testing of different languages are shown in Table (1) and (2) correspondingly.

Table 1. Statistics of training data

Language	Speakers	Lengths of Sentences	Total Duration of Training Samples	No. of Training sentences
Hindi	27 speakers, 23 male and 4 female	2-5 sec	440 sec	135
Telugu	24 speakers 20 male and 4 female	3-8 sec	440 sec	98
Gujarati	22 speakers 18 male and 4 female	2-7 sec	472 sec	132
English	25 speakers 24 male and 11 female	2-9 sec	420 sec	138

First of all, the speech files are hand-segmented to remove silence regions. with the help of WAVE-PAD software. Then speech is segmented into frames of length 23 msec (256 samples) and the overlapping between two frame was taken half of the frame length which is 11.5 msec (128 samples). Hamming window is used for smoothing. Then 12-dimensional Mel Frequency Cepstral Coefficients (MFCC) are extracted for each frame and were augmented in their time context. After taking MFCC its Delta and Delta-Delta Cepstral Coefficients are also extracted. The window length for delta and Delta-Delta Coefficients is K=9 and K=5 respectively. Cepstral Mean Subtraction (CMS) is applied to remove the effect of convoluting noises.

Separate GMM is used for each of the coefficient stream(MFCC, its Delta and Delta-Delta) for each language. Number of components in each GMM are kept 40. Now, for every language there are three GMMs, one each corresponding to different feature stream.

Table 2. Statistics of test data

Language	Speakers	Lengths of Sentences	No. of Test utterances
Hindi	35 speakers, 31 male and 4 female	2-5 sec	105
Telugu	22 speakers 18 male and 4 female	3-9 sec	62
Gujarati	22 speakers 18 male and 4 female	2-10 sec	88
English	28 speakers 14 male and 14 female	2-10 sec	91

Table 3. Performance comparisons for LID using simple EM and SMEM

Languages taken	Simple EM	SMEM	Efficiency gained
Hindi, English, Gujarati, Telugu	81.20 %	82.65 %	1.45 %
Hindi, English, Gujarati	85.21 %	85.21 %	0.00 %
Hindi, English, Telugu	84.70 %	86.20 %	1.50 %
Hindi, Telugu, Gujarati	80.78 %	81.96 %	1.18 %
English, Telugu, Gujarati	87.96 %	90.87 %	2.91 %
Hindi, English	91.26 %	92.72 %	1.46 %
Hindi, Gujarati	87.05 %	85.50 %	-1.55 %
Hindi, Telugu	91.02 %	91.62 %	0.60 %
English, Gujarati	93.85 %	94.41 %	0.56 %
English, Telugu	98.69 %	98.69 %	0.00 %
Telugu, Gujarati	86.67 %	91.33 %	4.66 %

In the first experiment all GMMs are trained using EM algorithm. Next we apply the split and merge algorithm and perform the usual EM iteratively until the log-likelihood is increasing. The log-likelihood is given by

$$\mathcal{L}(\{x_n, y_n, z_n\}|\Theta_l^x, \Theta_l^y, \Theta_l^z) = \sum_1^N \big[a * logp(x_n|\Theta_l^x) \\ +b * logp(y_n|\Theta_l^y) + c * logp(z_n|\Theta_l^z)\big], \quad (5)$$

where Θ_l^x are the parameters of GMM modeled using MFCC for language l, Θ_l^y are the parameters of GMM using delta Cepstral coefficients for language l and Θ_l^z are the parameters of GMM using delta-delta Cepstral coefficients for language l. x_n, y_n, z_n are Cepstral coefficients, delta Cepstral coefficients and delta-delta Cepstral coefficients correspondingly. It is assumed that these three streams are jointly statistically independent of each other. The maximum likelihood classifier hypothesizes i as the language of the unknown utterance, where

$$i = argmax_l \big[\mathcal{L}(\{x_n, y_n, z_n\}|\Theta_l^x, \Theta_l^y, \Theta_l^z)\big] \tag{6}$$

Table(3) shows the LID performance for both using simple EM and split and merge EM. The test is performed for values $a = 0.6, b = 10, c = 10$. These values of $a,\ b\ and\ c$ are approximated by experiments for which the performance is better. From the comparison results shown in Table(3) it is clear that SMEM outperform simple EM algorithm and gives better performance for LID.

5 Conclusions

A Split and Merge EM algorithm based approach is proposed to solve the language identification problem by using Gaussian mixture models.The problem of local maxima occurs in a mixture model is avoided by this split and merge EM (SMEM) algorithm. SMEM algorithm changes the parameters of some GMM components by split and merge operations. It improves the distribution of Gaussian components in the space which in-turn increases the log-likelihood of observing the data. This makes GMMs to learn the languages better in comparison to using the simple EM algorithm.

References

1. Muthusamy, Y.K.: A Segmental Approach to Automatic Language Identification. PhD thesis, Oregon Graduate Institute (1993)
2. Zissman, M.A., Singer, E.: Automatic language identification of telephone speech messages using phoneme recognition and N-GRAM modeling. In: Proc. ICASSP 1994, Adelaide, Austrailia (1994)
3. Nagrajan, T., Murthy, H.A.: Language identification using parallel syllable like unit recognition. In: Proc. ICASSP (2004)
4. Mary, L., Yegnanarayana, B.: Autoassociative Neural Network Models for Language ldentification. In: Proc. IClSlP (2004)
5. Hegde, R.M., Murthy, H.A.: Automatic Language Identification and Discrimination Using the Modified Group Delay Feature. In: Proc. ICISIP (2005)
6. Bimbot, F.E., Magrin-chagnolleau, I., Dutat, M.: Language recognition using time-frequency principal component analysis and acoustic modeling (2000)
7. Bilmes, J.A.: A gentle tutorial of the EM algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. Technical Report tr-97-021, International Computer Science Institute, Berkeley, California, USA (1997)
8. Ueda, N., Nakano, R., Ghabramani, Z., Hinton, G.E.: SMEM algorithm for mixture models. Neural Computation (2000)
9. Cheng, S.S., Wang, H.M., Fu, H.C.: A model-selection-based self-splitting gaussian mixture learning with application to speaker identification. In: EURASIP (2004)
10. Ormoneit, D., Tresp, V.: Improved gaussian mixture density estimates using bayesian penalty terms and network averaging. In: NIPS (1995)
11. Zhang, Z., Chibiao Chen, J.S., Chan, K.L.: EM algorithms for gaussian mixtures with split-and-merge operation. Pattern Recognition (2003)