Discriminative MLE training using a product of Gaussian Likelihoods

T. Nagarajan and Douglas O'Shaughnessy

INRS-EMT, University of Quebec
Montreal, Canada

raju,dougo@emt.inrs.ca

Abstract

In this paper, we describe a discriminative technique to determine an optimal HMM topology for the each of the models in a continuous speech recognition system such that the word error rate (WER) is minimized. In conventional model selection techniques such as Bayesian information criterion (BIC), the model complexity is determined without considering the other classes in a system. In our work, an optimal model topology is selected by considering how well a given model can discriminate examples of other classes from its own. By doing so, the estimated model parameters indirectly make sure that class separability is increased. In an earlier work [1], we have proposed this technique and experiments were carried out on an E-set. Presently, we extend it for building a syllable-based continuous speech recognition system. Preliminary experiments carried out on the TIMIT corpus show that a considerable reduction in WER can be achieved using the proposed technique over the BIC-based technique for model selection.

Index Terms: speech recognition, discriminative training, product of Gaussian.

1. Introduction

The most popular training method for a hidden Markov model (HMM) is maximum likelihood estimation (MLE). In MLE, with a pre-defined structure of a model and available training data, parameters of a HMM can be efficiently estimated [2] by maximizing the likelihoods of the training data. It is widely used because of the simplicity of its implementation using the Baum-Welch reestimation algorithm and its minimum variance property. However, the weakness of the MLE algorithm lies in the fact that the model parameters are estimated in isolation, in the sense that information about other classes in a given task is not considered. Apart from this, defining a structure of a HMM (especially, the number of states and mixtures) is one of the major issues in acoustic modeling.

In a speech context, the number of states is usually chosen based on either the number of acoustic variations that one may expect across the utterance or the length of the utterance, and the number of mixtures per state can be chosen based on the amount of training data available. A better alternative technique to choose the number of components of a model (number of states and mixtures) is the Bayesian information criterion (BIC). Recently, the BIC has been successfully used as a model selection technique in various classification tasks. In a speaker indexing task, the BIC has been used to switch between Vector Quantization (VQ) and Gaussian mixture model (GMM) based speaker models, and to choose the number of mixtures in a GMM [3]. In speech recognition, the BIC has been used to choose the number of mixture components, a covariance model [4], and the number of HMM states ([4], [5]). In handwriting recognition [6], the BIC is followed to choose the number of states of a HMM for different characters to be recognized. Even though the BIC is an efficient technique for controlling the complexity of a system, it does not consider information about other classes. This may lead to an increased error rate especially when dealing with most competitive and closely resembling other classes.

A discriminant measure-based technique was proposed in [7] for model complexity adaptation. In [7], for each class, a discriminant measure is computed by considering the corresponding class model and a fixed number of confusable models in a system. Based on this measure a decision is made either to increase or decrease the number of components of a model. Discriminative training techniques, such as Maximum mutual information (MMI) or Minimum classification error (MCE) based techniques, also consider the other models in a system to increase the likelihood of a given model. In MMI- or MCE-based techniques, the model parameters are updated directly instead of modifying the complexity of a model. In these discriminative techniques, either the parameters or the complexities of the models are optimized by considering the rest of the models or at least a subset of the most confusable models in the system.

In this paper, we propose a technique for complexity adaptation, which is similar to the technique presented in [7]. However, the major difference lies in the fact that the complexity of a model is adjusted by considering only the training examples of other classes and not their corresponding models. Hence, the entire training process is independent of the other models considered in a system. In this technique, the focus is given to how well a given model can discriminate training data of different classes, instead of considering how well training data is fitted with a correct model when compared to the other models in the system. If each of the models in a system can discriminate its own training examples from the others, it will indirectly make sure that the class separability will be increased. Even though it does not guarantee that the different classes will be well separated in a classification domain, it at least ensures that the resultant models are better than the BIC-based models.

In an earlier work [1], we have proposed this technique and experiments were carried out on an E-set. In the present work, the same technique is used to develop a syllable-based continuous speech recognition system. Here the discriminative optimization technique is applied for selecting a proper topology for each of the syllable models. The interest is shown only for reducing the word error rate and not for reducing the complexity of a model.

The rest of the paper is organized as follows. We first describe the model selection techniques used for analysis, namely, the BIC-

based technique and the proposed technique in Section 2. Under the proposed technique, we detail the importance of separating two classes in the likelihood space of a given model, and define an objective function to be minimized. In Section 3, the experimental set up used for the present study is described briefly, followed by a description on PoG-based model selection algorithm in Section 4. The performance of a syllable-based continuous speech recognition system trained using the proposed method is compared with a system trained using BIC and analyzed in Section 5.

2. Model selection Methods

As mentioned earlier, with a pre-defined structure of a model and the available training data, the parameters of a HMM can be efficiently estimated using the MLE algorithm. For our work, acoustic modeling is carried out using the MLE algorithm only. The focus is given to choosing a proper topology for the models to be generated, especially the number of mixtures per state. We compare the performance of the proposed technique with that of the conventional BIC-based technique. These systems are implemented using the Hidden Markov Model Toolkit (HTK). The details of the techniques used in the previous work [1] are discussed again for clarity purposes.

Let us consider the utterances of a class C_i as s_k^i, $k = 1, 2, ..., K_i$, where K_i is the number of training examples available for the class C_i. Let λ_i^m be the acoustic models of the class C_i, where m is the number of mixtures per state, which varies from $1, 2, ..., M$. Let the acoustic-likelihoods of the utterances of the class C_i for the given model λ_i^m be $p(s_k^i|\lambda_i^m)$.

2.1. Bayesian information criterion

In BIC-based complexity adaptation methods, the number of components of a model is chosen by maximizing an objective function that is essentially the likelihood of the training examples of a model penalized by the number of components in that model and the number of training examples. For the class C_i, let us assume that M models are pre-generated, each with a different number of mixtures per state. According to the BIC, the optimal model (λ_i^*) is the one which maximizes an objective function, as given below.

$$\lambda_i^* = \arg \max_{m=1,2,...,M} \log p(s_k^i|\lambda_i^m) - \alpha \frac{1}{2} (m\, S_i\, d)\, \log(K_i) \quad (1)$$

In Equation (1), S_i is the number of states in the model λ_i^m, d is the dimension of the feature vector, and α is an additional penalty factor used to control the complexities of the resultant models. In our experiments, we have varied the value of α and the results are reported for different amounts of complexities.

2.2. Product of Gaussian likelihoods (PoG)

Let us consider the utterances of two different classes (C_i and C_j) as s_k^i and s_k^j. Let λ_i and λ_j be the models of the classes, C_i and C_j, respectively. Let the likelihoods of the utterances of the class C_i for the given model λ_i be $p(s_k^i|\lambda_i)$. We can assume that these likelihoods are distributed normally in likelihood space with parameters μ_{ii} and σ_{ii}^2. Let this Gaussian be $N_{ii}(\mu_{ii}, \sigma_{ii}^2)$. The likelihoods of the utterances of the class C_j for the acoustic model of C_i are $p(s_k^j|\lambda_i)$ and are distributed with parameters μ_{ij} and σ_{ij}^2. Let this Gaussian be $N_{ij}(\mu_{ij}, \sigma_{ij}^2)$. If these two Gaussians overlap in likelihood space as shown in Figure 1(a), then to avoid errors during classification, the Gaussian likelihoods for the other

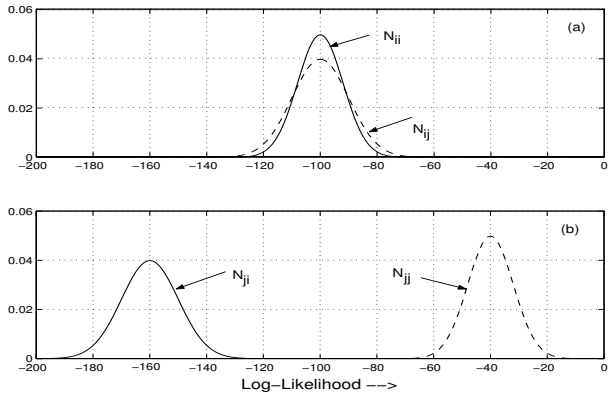

Figure 1: *Necessity to separate the Gaussian likelihoods. (a) The likelihood distributions of the utterances of the classes C_i and C_j for the given model λ_i. (b) The likelihood distributions of the utterances of the classes C_i and C_j for the given model λ_j.*

model (say N_{ji} and N_{jj}) should be well separated as in Figure 1(b). However separating N_{ji} and N_{jj} as in Figure 1(b) is possible only when the model λ_j is well trained. In other words, during training of the model λ_j, the acoustic likelihoods of all the utterances of the C_j should be maximized to a greater extent. To maximize the likelihood on the training data, the estimation procedure often tries to make the variances of all the mixture components very small. Although this leads to good training likelihood scores, it often provides poor matches to independent test data. This is especially true with the models of acoustically similar classes. To avoid this, it is always better to reduce the overlap between the likelihood Gaussians (say, N_{ii} and N_{ij}) of utterances of different classes (say, C_i and C_j) for a given model (say, λ_i).

In our case, we assume that two Gaussians overlap with each other considerably if either of the following conditions is met:

- If $\mu_{ii} \approx \mu_{ij}$, irrespective of their corresponding variances.
- If σ_{ii} or σ_{ij} is wide enough so that both the Gaussians overlap considerably.

In order to quantify the amount of overlap between two Gaussians, we can use error bounds, like the Chernoff or Bhattacharyya bounds. However, these error bounds are sensitive to the variances of the Gaussians. For example, if we consider the Bhattacharyya bound for error probability, even if $\mu_{ii} = \mu_{ij}$ and $\sigma_{ii} \neq \sigma_{ij}$, the error will not be equal to 1 (if prior probabilities are ignored). For our analysis, this may not be a suitable measure, at least when the variances of these two distributions are quite different. In [8], multiple probabilistic models of the same data are combined by multiplying the probabilities together. A similar logic is used here, but for estimating the amount of overlap between two Gaussians as described below. Let the product of $N_{ii}(\mu_{ii}, \sigma_{ii}^2)$ and $N_{ij}(\mu_{ij}, \sigma_{ij}^2)$ be $N_k(\mu_k, \sigma_k^2)$.[1] This can be written as

$$\begin{aligned} N_k(\mu_k, \sigma_k^2) &= N_{ii}(\mu_{ii}, \sigma_{ii}^2)\ .\ N_{ij}(\mu_{ij}, \sigma_{ij}^2) \quad (2) \\ &= K\, e^{-\left[\frac{(p(s_k^i|\lambda_i) - \mu_{ii})^2}{2\sigma_{ii}^2} + \frac{(p(s_k^j|\lambda_i) - \mu_{ij})^2}{2\sigma_{ij}^2}\right]} \end{aligned}$$

[1] In the present study, N_k is not normalized, as this will not affect its use in Equation (10).

where

$$K = \frac{1}{2\pi\sigma_{ii}\sigma_{ij}}. \quad (3)$$

For the product of the Gaussians, the mean (μ_k) can be given as

$$\mu_k = \frac{\sigma_{ij}^2\mu_{ii} + \sigma_{ii}^2\mu_{ij}}{\sigma_{ii}^2 + \sigma_{ij}^2}. \quad (4)$$

In order to quantify the amount of overlap between two different Gaussians, we define the following ratio ($\mathcal{O}$).

$$\begin{aligned}\mathcal{O} &= \frac{\max[N_{ii}(\mu_{ii}, \sigma_{ii}^2)\,.\,N_{ij}(\mu_{ij}, \sigma_{ij}^2)]}{\max[N_{ii}(\mu_{ii}, \sigma_{ii}^2)\,.\,N_{ii}(\mu_{ii}, \sigma_{ii}^2)]} \\ &= \frac{Nr}{Dr}. \end{aligned} \quad (5)$$

In Equation (5),

$$Nr = \frac{1}{2\pi\sigma_{ii}\sigma_{ij}}\; e^{-\left[\frac{(\mu_k-\mu_{ii})^2}{2\sigma_{ii}^2} + \frac{(\mu_k-\mu_{ij})^2}{2\sigma_{ij}^2}\right]} \quad (6)$$

$$\text{and}\;\; Dr = \frac{1}{2\pi\sigma_{ii}^2}. \quad (7)$$

From Equations (6) and (7), Equation (5) can be written as

$$\mathcal{O} = \frac{\sigma_{ii}}{\sigma_{ij}}\; e^{-\left[\frac{(\mu_k-\mu_{ii})^2}{2\sigma_{ii}^2} + \frac{(\mu_k-\mu_{ij})^2}{2\sigma_{ij}^2}\right]}. \quad (8)$$

If $\mu_{ii} = \mu_{ij}$, then Equation (8) reduces to

$$\mathcal{O} = \frac{\sigma_{ii}}{\sigma_{ij}}. \quad (9)$$

However, for this case we expect the overlap $\mathcal{O}$ to be equal to 1. To achieve this, Equation (8) is further normalized as given below.

$$\begin{aligned}\mathcal{O}_{\mathcal{N}} &= \mathcal{O}\frac{\sigma_{ij}}{\sigma_{ii}} \\ &= e^{-\left[\frac{(\mu_k-\mu_{ii})^2}{2\sigma_{ii}^2} + \frac{(\mu_k-\mu_{ij})^2}{2\sigma_{ij}^2}\right]}. \end{aligned} \quad (10)$$

The resultant $\mathcal{O}_{\mathcal{N}}$ is used as a measure to estimate the amount of overlap between two Gaussians. The steps followed during training are given below.

3. Experimental setup

For the present work, the TIMIT corpus is considered for both training and testing. The word-lexicon is created only with the test words. For the words that are in common in train and test data, pronunciation variations are taken from the transcriptions provided with the corpus and for the rest of the test words only one transcription is considered.

Syllabification software [9] available from NIST was used to extract the syllables from the phonetic segmentation boundaries given in the TIMIT corpus. For all the phonetic transcriptions available with the TIMIT corpus, the first-best results given by the syllabification algorithm are considered. Even though the number of unique syllables in the training corpus is $\approx$ 5000, most of the syllables have very few examples. In our work, we have considered 200 syllables that have more than 50 examples. The rest of the syllables are replaced by their corresponding phonemes in the transcriptions as described in [10]. 200 syllable and 46 monophone models are initialized using hand-labeled data available with the corpus. For initialized models, the number of states is fixed based on the number of phonemes for a given sub-word unit and the number of mixtures per state is considered as one. Here, the monophone models are generated in the context of syllables. For the re-estimation of model parameters, a standard Viterbi alignment procedure is used. The number of mixtures per state for each model is then increased to 30, in steps of 1, by a conventional mixture splitting procedure. Each time, the model parameters are re-estimated twice.

For the experiments, no optimization is carried out other than a word-insertion penalty optimization. Further, in order to clearly see the effect of syllabic units in combination with phone-sized units during Viterbi-alignment, no word level n-gram statistics are used. Testing is carried out on the whole test corpus of TIMIT.

4. PoG-based model selection

The steps followed for selection of a proper model using the information derived from the PoG are given below.

1. For the class C_i, extract all the corresponding models from the previously trained sets (refer section 3). Let these models be λ_i^m, where $m = 1, 2, ..., M$. As a first step, for the class C_i, a single mixture model (λ_i^1) is considered.
2. For the given model λ_i^m, compute the acoustic-likelihoods of the utterances of all the classes separately. Here, we assume that the likelihoods are distributed normally in the likelihood space. Let the distributions and their corresponding means and variances be N_{ij}, μ_{ij}, and σ_{ij} respectively, where $j = 1, 2, ..., N$.
3. Compute the $\mathcal{O}_{\mathcal{N}}$ for each pair 'N_{ii}, N_{ij}', where $j = 1, 2, ..., N$ and $j \neq i$.[2]
4. For any of the pairs, if $\mathcal{O}_{\mathcal{N}}$ is greater than ϵ, increase the number of mixtures per state by 1 ($m = m + 1$) and repeat the steps 2 to 4.
5. Otherwise, the corresponding model λ_i^m is considered as the optimal model.

In the above training process, if the value of ϵ is reduced, we can expect a better performance. However, below some value for ϵ the models may become over-trained. For performance analysis of the proposed technique for different amounts of overlap, the above training process is carried out for different values of ϵ and the results are reported in the next section. In this work, the model optimization technique is applied only to the syllable models. For all the phoneme models, the number of mixtures is fixed as 16 per state.

5. Performance analysis

The BIC and the PoG-based models for all the syllables are generated as explained in the previous section and used for testing the performance of the recognizer. In some cases, since the model size

[2]Here, N can be the same as the number of syllable models available in the system. However, to reduce the computation time, in this work, only a restricted ($N = 10$) number of syllables is considered. For each of the syllables, the N neighbours (closely located syllables in the acoustic space) are derived with the initialized model and the same N neighbours are used for the rest of the optimization procedure.

seems to grow uncontrollably, we fixed the maximum number of mixtures per state at 30. The performances of these two types of systems are given in Table 1 for comparison. For complexity analysis, the WER of these two systems with respect to the complexity is shown in Figure 2.

Table 1: Word error rate (in %) as a function of α in the BIC-based system and ϵ in the PoG-based system

BIC-based system			PoG-based system		
α	# Gaussians (X 10^3)	WER	ϵ	# Gaussians (X 10^3)	WER
2.00	452	48.5	.98	490	48.9
1.30	505	45.8	.90	807	45.0
1.00	564	44.3	.80	940	43.9
0.80	638	43.9	.70	1049	43.6
0.60	769	43.6	.60	1135	42.8
0.40	1081	44.4	.50	1219	42.9
0.20	1517	47.7	.40	1292	43.4
			.30	1364	45.8

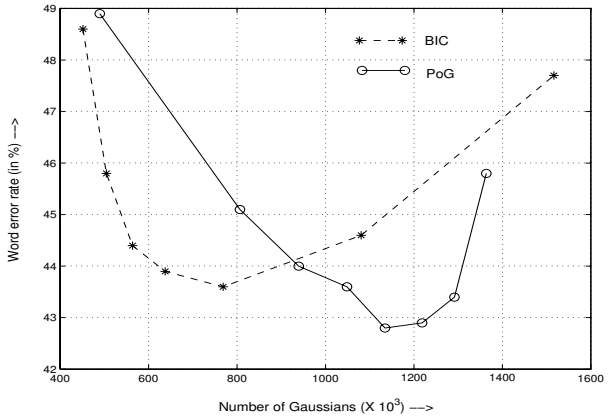

Figure 2: WER curves of the BIC (as a function of α) and the PoG (as a function of ϵ) based systems

From a comparative analysis, the following observations may be made.

- For lower complexities, the error rates of the BIC-based system are found to be lower when compared to that of the PoG-based system.
- For higher complexities, the PoG-based system clearly dominates and the minimum error rate is achieved by the PoG-based system.
- In the case of the PoG-based system, one can observe that, as the overlap reduces, the error rate also reduces and for extreme cases, since the models are over-trained, the error rate again increases. This analysis shows that one can fix the value for ϵ as 0.4 to 0.8 for better performance.

From these observations, we may conclude that if the WER is of primary importance then the PoG-based system can be preferred.

6. Conclusions

In conventional techniques for model optimization, the topology of a model is optimized either without considering other classes (as in BIC), or considering a subset of competing models (as in discriminative techniques). In our work, we have made an attempt to optimize the topology of the models by considering whether a given model can discriminate training utterances of other classes from its own. The major advantage in our technique is that the optimization is carried out discriminatively and at the same time, independent of the other models available in a system. Even though the complexity of the system seems to grow as the amount of overlap between different classes in the likelihood-space is reduced, it ensures a lower WER. In the present study, we concentrated only on modifying the number of mixtures of a model. This can be extended to optimally decide the number of states also. Further, instead of a uniform number of mixtures for all the states, varying numbers of mixtures can be tried, which may reduce the complexity of the system.

7. References

[1] T. Nagarajan and D. O'Shaughnessy, "Discriminative optimization of HMM topology using product of likelihood gaussians (under review)," *IEEE Signal Processing Letters*.

[2] L. R. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition*. New Jersey: Prentice Hall, 1993.

[3] M. Nishida and T. Kawahara, "Speaker model selection based on the Bayesian information criterion applied to unsupervised speaker indexing," *IEEE Trans. Speech and Audio Processing*, vol. 13, no. 4, pp. 583–592, July 2005.

[4] S. S. Chen and R. A. Gopinath, "Model selection in acoustic modeling," in *Eurospeech*, 1999, pp. 1087–1090.

[5] W. Chou and W. Reichl, "Decision tree state tying based on penalised Bayesian information criterion," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 1, Mar. 1999, pp. 345–348.

[6] D. Li, A. Biem, and J. Subrahmonia, "HMM topology optimization for handwriting recognition," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 3, May 2001, pp. 1521–1524.

[7] M. Padmanabhan and L. R. Bahl, "Model complexity adaptation using a discriminant measure," *IEEE Trans. Speech and Audio Processing*, vol. 8, no. 2, pp. 205–208, Mar. 2000.

[8] S. S. Airey and M. J. F. Gales, "Product of gaussians and multiple stream systems," in *Proceedings of IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 1, May 2001, pp. 497–500.

[9] B. Fisher, "tsylb2-1.1 - syllabification software," http://www/nist.gov/speech/tools, August 1996.

[10] A. Ganapathiraju, J. Hamaker, J. Picone, M. Ordowski, and G. R. Doddington, "Syllable-based large vocabulary continuous speech recognition," *IEEE Trans. Speech and Audio Processing*, vol. 9, no. 4, pp. 358–366, May 2001.