Optimal model order selection based on regression tree in speaker identification

Shilei Zhang*, Junmei Bai*, Shuwu Zhang*, Bo Xu*+

High Technology Innovation Center*; National Laboratory of Pattern Recognition+
Institute of Automation, Chinese Academy of Sciences, Beijing, 100080
{slzhang, jmbai, swzhang, xubo}@hitic.ia.ac.cn

Abstract
In this paper, we propose a new method to select the optimal model order for the initialization of Gaussian Mixture speaker Models (GMM) based on regression tree in text-independent speaker identification system. The objective is to choose the optimal number of components which is necessary to adequately model a speaker for a good speaker identification performance according to the Bayesian Information Criterion (BIC) and agglomerative clustering. One obvious advantage of such method is that it provides a flexible framework to select an optimal speaker model order based on the training data for each client speaker. The experimental results on the YOHO corpus show that adaptive model mixture components achieves better performance, especially considering the fact that different speakers have different amounts of available enrollment data.

1. Introduction
Speaker Identification (SID) is the process of automatically recognizing who is speaking by using speaker-specific information included in speech waves. GMM has been successfully applied to the tasks of text-independent SID.

Given training speech from a speaker, the goal of speaker model training is to estimate the parameters of the GMM, which in some sense best matches the distribution of the training feature vectors. By far the most popular and well-established method is maximum likelihood (ML) estimation [1]. However, ML parameter estimates can be obtained iteratively using a special case of the expectation-maximization (EM) algorithm [2]. Two critical factors in training a Gaussian mixture speaker model are selecting the order of the mixture and initializing the model parameters prior to the EM algorithm. The EM algorithm is guaranteed to find a local maximum likelihood model regardless of the starting point, but the likelihood equation for a GMM has several local maxima corresponding to different starting models. There are no good theoretical means to guide one in either of these selections, so they are best experimentally determined for a given task [1].

However, it is unreasonable to believe that the speaker identification information is identically involved in the same amount of enrollment data. Especially, when we can not get the same amounts of data for different speakers in domains such as commercial security and criminological examination, it is not feasible to apply the model with the same mixture number. To solve the problem, we present a new tree-based model order selection method. It tries to achieve accurate order for every speaker model via data given algorithm based on regression tree. In other words, we build a regression tree using LBG algorithm and K-means algorithm [3][4], then select the number of Gaussian distributions based on the BIC [5][6] according to the training data, finally we will agglomerate nodes with sparse data [7]. Detailed experimental results on YOHO corpus indicate that the proposed approach outperforms the conventional model initialization algorithm, especially when the variation of amount of available enrollment data for client speakers is large.

The rest of the paper is organized as follows: In section 2 we review the identification system. Section 3 describes the proposed regression tree procedure. In section 4 experiments on YOHO data are presented and the results will be discussed in section 5. In Section 6 we will draw some experimental conclusions.

2. Speaker identification system review

Figure 1: Typical Identification System.

The basis for identification system is the Gaussian mixture models which can be used to represent speakers. More specifically, the distribution of feature vectors extracted from a person’s speech is modeled by a Gaussian mixture density. For a feature vector denoted as x, the mixture density for speaker s is defined as:

$$p(x | \lambda_s) = \sum_{i=1}^{M} p_i b_i(x)$$  \hspace{1cm} (1)

The density is a weighted linear combination of M component unimodal Gaussian density $b_i(x)$, each parameterized by a mean vector $\mu_i$ and covariance matrix $\Sigma_i$. Collectively, the parameters of a speaker’s density model are denoted by
Maximum likelihood estimates of the model parameters are obtained using the EM algorithm.

Typical identification system is represented by Fig. 1. For an utterance \( X = \{x_1, \ldots, x_T\} \) and a reference group of \( S \) speakers represented by models \( \{\lambda_1, \lambda_2, \ldots, \lambda_S\} \), identification is performed by computing the maximum likelihood classification rule:

\[
\hat{s} = \arg \max_{1 \leq s \leq S} p(X | \lambda_s)
\]  

(2)

The speaker whose model produces the largest score is then determined to be the identified speaker.

3. Optimal model order selection and mode initialization

Model training consists of two steps, the first step is to estimate initial parameters for each model of each speaker from the training data. The second step makes better estimates of the model parameters creating the models to be used in identification. In order to select the optimal model order and initialize the model parameters for the given data, we present the regression tree method.

Regression trees are attractive due to their simplicity in model interpretation, and are particularly suited for effective data mining [7]. One of the important attributes of tree based regression is its capability to generalize input-output mapping from the limited set of training examples. A regression tree is a binary tree constructed by repeatedly splitting (sub)sets of learning cases into two descendant subsets. Each node of a tree contains a subset of cases. A node that does not have descendant nodes is a terminal node. The root node comprises the entire samples. The left and right child nodes contain disjoint subsets of the parent content and are defined by splitting the parent node. Fig. 2 illustrates the algorithm to find the optimal model order and initial model for a given task. Each of these steps will be discussed in more detail in the following sections.

3.1. Regression tree building

In this paper, the regression tree is built by using LBG algorithm and K-means algorithm. The algorithm is formally implemented by the following recursive procedure:

1. Initiation: Design a 1-vector codebook as the root node; this is the centroid of the entire set of training vectors.

2. Splitting: Double the size of codebook by splitting each current codebook (parent node) \( y_n \) according to the rule (3), where \( n \) varies from 1 to the current size of codebook, and \( \varepsilon \) is a splitting parameter:

\[
y_n^+ = y_n(1+\varepsilon) \quad ; \quad y_n^- = y_n(1-\varepsilon)
\]  

(3)

3. Clustering: Beginning with the new codebook, split each parent node into two child nodes using K-means clustering algorithm described in the following recursive process.

1) Nearest-Neighbor Search: For each training vector, find the codeword in the current codebook that is closest in terms of similarity measurement, and assign that vector to the corresponding node.

2) Mean update: Update the mean in each node using the centroid of the training vectors assigned to that node.

3) Repeat steps 1) and 2) until the average distance falls below a present threshold.

4. Repeat step 2, 3 until the stopping criteria are met.

Two aspects should be pointed out for the algorithm. First, in step 3, we assign the vector to the subset in terms of Mahalanobis distance that takes into account not only the average value but also its variance and the covariance of the variables measured. Secondly, as mentioned in step 4, the tree building process goes on until some criteria are met. The process is stopped: (1) there is only one observation in each of the child nodes; (2) the farthest distance among all observations within each child node falls below a present threshold; or (3) the process has reached the limit on the number of levels in the maximal tree predefined according to the given task.

Now it is reasonable to assume that the regression tree corresponds to the acoustic space of the enrollment data. Each node is assigned an acoustic class (i.e. a unimodal Gaussian density) represented by the mean vector, covariance matrix and mixture weight, while every level can be viewed as an initial GMM. Next we will select the optimal model order and initialize the speaker model from the tree.

![Figure 2: The process of initializing model.](image)
3.2. Tree level selection

As mentioned above, the deep level models contain too many components, which will reduce performance when there are a large number of model parameters relative to the available training data and can also result in excessive computational complexity. The few mixture components can produce a speaker model which does not accurately model the distinguishing characteristics of a speaker’s distribution. In this step, we will select the number of Gaussian distributions contained within a certain level based on the BIC according to the training data. The BIC of the GMM for a speaker s is formulated with the following function (4), where \( \log P(X | \lambda) \) is a log likelihood of the training data X by the GMM when the number of mixtures is M, \( d \) is the dimension of the acoustic feature, N is the number of frames of the training data.

\[
BIC = \log P(X | \lambda) - \frac{1}{2} M (2d + 1) \log N \tag{4}
\]

3.3. Agglomeration

When the observations of a node within selected level are sparse, we will merge the nodes. The similarity measure between the nodes is the symmetric Kullback-Leibler divergence [8]. This measure is defined for measuring the distance between two given Gaussian probability density functions, \( f \) and \( g \), as:

\[
d(f, g) = \int f(x) \log \frac{f(x)}{g(x)} dx + \int g(x) \log \frac{g(x)}{f(x)} dx \tag{5}
\]

3.4. Initializing model parameters

Finally, we obtain a group of clusters and all the training vectors are assigned to a particular cluster. The number of clusters is optimal model order and the initial model can be parameterized by the mean vectors, covariance matrices and mixture weights from all component cluster densities. The mean of a cluster can be obtained by calculating the mean of all the vectors assigned to that particular cluster. The covariance matrix can be obtained by computing the covariance matrix of the assigned vectors. The mixture weights can be found out by calculating the proportion of the number of vectors assigned to a particular cluster.

4. Experimental results

4.1. Database and experimental condition

The accuracy and efficiency of the proposed method is evaluated by the identification task based on YOHO database [9]. The data consists of 138 speakers (106 male, 32 female) recorded in an office setting over a span of 3 months. The enrollment data consist of four sessions each containing 24 utterances while the identification data has 10 sessions of 4 utterances each. All of the test data are used for the following experiments.

During model training, the speech was preemphasised using a first-order finite impulse response (FIR) filter with a factor of 0.97. Silence was removed by discarding low energy frames using an energy-based speech activity detection algorithm. The speech was parameterized every 12 ms from 24 ms overlapping windows. Each frame was parameterized by a vector consisting of 16 melfrequency cepstral coefficients (MFCC) and normalized log frame energy as well as its first order derivatives.

The following three sets of experiments are conducted in order to compare the performance of conventional model initialization method with tree-based model selection method using the same amounts and the different amounts of training data, respectively. No significant difference was found among the conventional initial models with the same model order [1], so the rest of experiments in this paper only uses K-means clustering algorithm to initialize means, variances and mixture weights as the conventional method.

4.2. Training model with sufficient enrollment data

All four enrollment sessions are used in this section. Fig. 3 shows the percent correct identification performance versus the number of Gaussian components of speaker model initializing by K-means method. Performance peaks at 512 components. We can observe that the proposed regression tree method improves performance but with a slight increment from Table 1.

4.3. Training model with insufficient enrollment data

Only the first session is used for training in this section. Fig. 3 displays performance peaks at 64 components. We can see that tree-based method has some improvement over K-means from Table 1. However, the best mixture number is obtained by so many experiments using K-means algorithm. It is also evident that model order selection becomes more effective with smaller amounts of training data.

4.4. Training model with variant enrollment data

To gain some understanding of how performance varies with the different amounts of test data, all speakers are partitioned into 4 subsets, which sequentially builds up speaker models using only session 1, sessions 1-2, sessions 1-3 and all of the four enrollment sessions. Performance peaks at 32 components.

### Table 1: Identification performance comparison of model initialization methods

<table>
<thead>
<tr>
<th></th>
<th>K-means (peak)</th>
<th>Tree-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>The same amounts of data</td>
<td>98.85</td>
<td>98.89</td>
</tr>
<tr>
<td>The different amounts of data</td>
<td>93.28</td>
<td>96.39</td>
</tr>
</tbody>
</table>

![Figure 3: Speaker identification performance comparison of model initialization methods](image-url)
From Table 1, we can see when building models using the different amounts of data for each speaker, the performance of proposed method provides a 3.1% absolute improvement. Under such condition, the uniform model order will lose some speaker recognizable information involved in the training data.

4.5. Speakers versus model orders

Fig. 4 and 5 show the client speakers versus the optimal model orders obtained from the regression tree. A visual illustration of the distribution of model order for every speaker is given. We can see the proposed method can adaptively select optimal model order according to the amounts of training data. Compare with the traditional modeling methods, the tree-based method provides a flexible and feasible way to automatically optimize the initial model.

5. Discussion

From the experimental results described in the last section, the proposed tree-based model initialization algorithm can converge to an initial model which reflects the recognizability of different speakers included in training data. There is significant improvement in the SID performance when the adaptive optimal model order is used for each speaker.

For a given task, when the new training data are provided even if the amounts are the same, we must determine the model order over again by many experiments using the conventional algorithm. However, for the proposed method, the model order is based on the change in the amounts of enrollment data for a client speaker. Therefore, we can obtain the optimal model order for different tasks just by the proposed method directly. For the various training data, regression tree model initialization algorithm is more feasible and more available than the uniform model order initialization method.

Here, we should point out the obvious advantage of such method is that it can avoid repeated experiments and converge to an optimal model order for a particular client speaker.

6. Conclusions

In this paper, a data-driven model initialization algorithm based on regression tree for speaker modeling has been presented in SID system. The method can select more accurate model order for the given data for each speaker. Experimental results show that tree-based algorithm has better performance than conventional uniform model approaches, especially when the amounts of available enrollment data are different to the client speakers.

7. Acknowledgements

This work was partially supported by the National Natural Science Foundation of China (NSFC) under the grant No. 60475014.

8. References


