Speaker Clustering Using Direct Maximization of A BIC-based Score

Wei-Ho Tsai

Department of Electronic Engineering & Graduate Institute of Computer and Communication Engineering, National Taipei University of Technology, Taipei, Taiwan

whtsai@ntut.edu.tw

Abstract

This paper presents an effective method for clustering unknown speech utterances based on their associated speakers. The proposed method jointly optimizes the generated clusters and the required number of clusters according to a Bayesian information criterion (BIC). The criterion assesses a partitioning of utterances based on how high the level of within-cluster homogeneity can be achieved at the expense of increasing the number of clusters. Unlike the existing methods, in which BIC is used only to determine the optimal number of clusters, the proposed method uses BIC in conjunction with a genetic algorithm to determine the optimal cluster where each utterance should be located. The experimental results show that the proposed speaker-clustering method outperforms the conventional methods.

Index Terms: speaker clustering, Bayesian information criterion, genetic algorithm, divergency

1. Introduction

In addition to transcribing the linguistic meanings behind speech, determining “who spoken when” [1,2] in an audio stream is also a topic of great interest in the research field of spoken document retrieval. This topic involves two problems in general: segmenting an audio recording into speech utterances that contain only one speaker’s voice [3,4], and grouping utterances from the same speaker into a cluster [5-9].

In this paper, we concentrate on the latter problem, referred to as speaker clustering. Given N speech utterances, each of which is assumed from one of the P unknown speakers, where N ≥ P, the aim of speaker clustering is to partition the N utterances into M clusters, such that M = P, and each cluster consists of utterances from only one speaker.

Currently, most speaker-clustering methods use a hierarchical agglomerative clustering (HAC) framework [3-7] to group together similar utterances in a sequential manner, and then determine how many groups should be generated using the Bayesian information criterion (BIC) [4,6]. However, though HAC attempts to make the voice characteristics within a newly generated cluster as homogeneous as possible, it does not guarantee that the homogeneity for all the clusters can be summed to reach a maximum, since its decision does not consider the interaction between the new cluster to be generated and existing clusters. As a result, some mis-clustering errors, arising from grouping different-speaker utterances together can propagate down the whole process, and hence limit the clustering performance. In addition, the process of generating clusters is virtually independent of the determination of the optimal number of clusters, and the latter trusts the former completely, the inevitable errors from the former can propagate to the latter, which may lead to a poor estimation of the speaker population size.

To improve the performance of speaker clustering, [8] and [9] develop methods of maximizing the within-cluster homogeneity of speaker voice characteristics in a global fashion, instead of in a cluster-by-cluster manner used in HAC. However, both [8] and [9] still follow the principle of the conventional BIC-based method for estimating the speaker population size, which is performed separately from the generation of clusters; hence, they may suffer from the problem of error propagation. In this study, we attempt to overcome the limitations of the above-mentioned methods by jointly optimizing the generated clusters and the required number of clusters with BIC. In contrast to the conventional methods, in which BIC is used only to determine the optimal number of clusters, the proposed method uses BIC to determine both the optimal number of clusters and the optimal cluster where each utterance should be located. This is done by scoring various partitionings of utterances, based on how homogeneous the utterances within a cluster can be achieved and how small the number of clusters needs to be generated. The partitioning which produces the largest BIC-based score would approach optimum in terms of both the cluster homogeneity and size. To search for such a partitioning, we develop methods for characterizing the cluster homogeneity, representing the BIC-based scores, and maximizing the BIC-based scores.

2. Maximum BIC Clustering (MBICC)

In general, the greater the number of clusters we generate, the higher the level of speaker homogeneity within each cluster we can obtain. However, if we generate too many clusters, a single speaker’s utterances would be split across multiple clusters, and hence the speaker clustering will not be completed. Clearly, the optimal number of clusters is equal to the speaker population size, which is, however, unknown and needs to be estimated. In attempts to optimize the clustering, the best solution would be to maximize the level of speaker homogeneity within each cluster while minimize the number of generated clusters. We therefore try to design a score for assessing a partitioning of the utterances based on how high the level of within-cluster homogeneity can be achieved at the expense of increasing the number of clusters. Then, speaker clustering can be solved by optimizing the score.

2.1. Scoring the partitionings via the BIC

Our basic strategy for scoring a partitioning of utterances is based on the Bayesian information criterion (BIC) [10]. The BIC is a model selection criterion, which evaluates a parametric model on the basis of how well the model fits a data set, and how simple the model is. If there are K parametric models, $L_1, L_2, \ldots, L_K$ built for characterizing a data set $O$, then each model is scored via a BIC value:

$$BIC(L_i) = \log Pr(O | L_i) - 0.5 \gamma \#(L_i) \log |O|,$$

(1)
where $\text{Pr}(O|\Lambda)$ is the likelihood probability represents how $\Lambda$ fits $O$, $\#(\Lambda)$ denotes the number of free parameters in model $\Lambda$, $|O|$ is the size of $O$, and $\gamma$ is a penalty factor. The criterion favors the model having the largest value of BIC. If we consider each of the possible partitionings of the utterances as a model for characterizing speaker information in the utterances, then the first term in the right side of Eq. (1) accounts for the overall with-cluster homogeneity, and the second term penalizes the partitioning with a large number of clusters. Accordingly, the best partitioning would be the one producing the largest value of BIC.

2.2. Modeling the partitionings

To compute Eq. (1) explicitly, the prerequisite is to perform various possible partitionings and then represent the result of each partitioning as a parametric model. Following [9], we characterize each possible partitioning as a set of indices $H = \{h_1, h_2, \ldots, h_N\}$ for the $N$ utterances to be clustered, where $h_n, 1 \leq n \leq N$, indicates the cluster where the $n$-th utterance should be located. Note that each index $h_n$ is an integer between 1 and $M$, and the value of $M$ is unknown and to be determined. Let $X = \{X_1, X_2, \ldots, X_J\}$ denote $N$ utterances to be clustered, each of which is represented by a frame-based spectral feature stream, i.e., $X_n = \{x_{n1}, x_{n2}, \ldots, x_{nT_n}\}$, with $T_n$ frames, $1 \leq n \leq N$. Then, a partitioning is deemed homogeneous if $h_m = h_l$ for any utterances $X_m$ and $X_l$ similar to each other, and $h_m \neq h_l$ for any $X_m$ and $X_l$ dissimilar to each other.

To quantify the speaker mixture homogeneity as a comparable value, we employ Gaussian mixture modeling technique [11], by virtue of its ability to capture the characteristics of speaker voice residing in the long-term spectrum. First, all the utterances are pooled together to form an utterance-independent Gaussian mixture model (GMM), which represents a generic voice characteristic independent of speaker and speech content. The parameters of the utterance-independent GMM are denoted by $\lambda = \{\mu_i, \Sigma_i, \pi_i\}, 1 \leq i \leq J\}$, where $J$ is the number of mixed Gaussian densities, $\mu_i$ are mixture weights, $\mu_i$ are mean vectors, and $\Sigma_i$ are covariance matrices. These parameters are estimated using the Expectation-Maximization (EM) algorithm [12]. After that, an utterance-dependent GMM $\lambda_n = \{\mu_{nj}, \Sigma_{nj}, \pi_{nj}\}, 1 \leq n \leq N\}$ is formed for each utterance $X_n, 1 \leq n \leq N$, which represents a specific speaker’s voice characteristics observed in $X_n$. The parameters of $\lambda_n$ are estimated by adapting $\lambda$ based on Maximum A Posteriori (MAP) [11]. Next, for each cluster $c_m, 1 \leq m \leq M$, we generate a cluster-dependent GMM $\lambda^{(m)} = \{w_{mj}, \mu_{jm}, \Sigma_{jm}\}, 1 \leq j \leq J\}$ using all the utterances assigned to $c_m$. The parameters of $\lambda^{(m)}$ are also estimated by adapting $\lambda$ based on MAP. Accordingly, if all the utterances within a cluster are from the same speaker, then each utterance-dependent GMM should be similar in some sense to the associated cluster-dependent GMM. Then, by characterizing the similarities between models with divergence [13], we can compute the speaker homogeneity for a partitioning $H$ using

$$
\log \text{Pr}(X | H) = \sum_{m=1}^{M} \sum_{n=1}^{N} \log S(\lambda^{(m)}, \lambda_n) \delta(h_n, m).
$$

where $S(\lambda^{(m)}, \lambda_n)$ denotes the divergency-based similarity between models $\lambda^{(m)}$ and $\lambda_n$ [14];

$$
S(\lambda^{(m)}, \lambda_n) = \sum_{j=1}^{J} w_{nj} \exp[-D(\mu_{jm}, \mu_{jn}; \Sigma_{jm}, \Sigma_{jn})],
$$

and $D(\mu_{jm}, \Sigma_j, \mu_{jn}, \Sigma_n) = 
\frac{1}{2} \left( \frac{1}{\Sigma_j} + \frac{1}{\Sigma_n} \right) \left( \mu_{jm} - \mu_{jn} \right)^T \left( \frac{1}{\Sigma_j} + \frac{1}{\Sigma_n} \right) \left( \mu_{jm} - \mu_{jn} \right) + \frac{1}{2} \text{Tr} \left( \frac{1}{\Sigma_j} \Sigma_j - \frac{1}{\Sigma_n} \Sigma_n \right) + \frac{1}{2} \text{Tr} \left( \Sigma_j \Sigma_n \right) - \log \left( \frac{1}{\Sigma_j} \right) - \log \left( \frac{1}{\Sigma_n} \right) - \frac{1}{2} \text{log} \frac{\text{det} \Sigma_j}{\text{det} \Sigma_n}.$$

is the divergence between Gaussian distributions $N(\mu_{jm}, \Sigma_j)$ and $N(\mu_{jn}, \Sigma_n)$. Tr() denotes the trace of a matrix, and $R$ is the dimension of the feature vectors. For greater computational efficiency, we keep the mixture weights unchanged during MAP adaptation, i.e., $w_{nj}^{(m)} = w_{nj}, 1 \leq j \leq J$. Since the mixture components of $\lambda^{(m)}$ and $\lambda_n$ are aligned, Eq. (3) can be simplified as

$$
S(\lambda^{(m)}, \lambda_n) = \sum_{j=1}^{J} w_{nj} \exp[-D(\mu_{jm}, \mu_{jn}; \Sigma_{jm}, \Sigma_{jn})].
$$

Consider the BIC in Eq. (1) for our clustering framework. The data set $O$ is the $N$ utterance-dependent GMMs, which is further “modeled” by $M$ cluster-dependent GMMs in accordance with $H$. Hence, the size of data are the number of utterance-dependent GMMs, i.e., $|O| = N$, which does not depend on the utterance duration. In addition, since the configuration of the data (utterance-dependent GMMs) are the same as that of the model (cluster-dependent GMMs), the number of free parameters in a model $\Lambda$ can be considered as the number of generated clusters, i.e., $\#(\Lambda) = M$, which is independent of the number of Gaussian densities used and the dimensionality of feature vectors. Accordingly, a score based on Eq. (1) for partitioning $H$ can be computed using

$$
B(H) = \sum_{n=1}^{N} \sum_{m=1}^{M} \log S(\lambda^{(m)}, \lambda_n) \delta(h_n, m) - 0.5 \gamma M \log N.
$$

Our goal is thus to find an optimal $H^*$, such that $B(H)$ is maximized, i.e.,

$$
H^* = \arg \max H B(H).
$$

2.3. Optimizing the BIC score via the genetic algorithm

Since the optimal number of clusters to be generated, $M$, is an unknown value between 1 and $N$, there are $N^N$ possible solutions of $H$ to Eq. (7). Thus, it may be costly prohibitive to perform exhaustive search, i.e., to examine all possible solutions and then determines the best one. To solve this problem, we apply the genetic algorithm (GA) [15] by using its global scope and parallel searching power.

The basic operation of the GA is to explore a given search space in parallel by means of iterative modifications of a population of chromosomes. Each chromosome, encoded as a string of alphabets or real numbers called genes, represents a potential solution to a given problem. In our task, a chromosome is exactly a legitimate $H$, and a gene corresponds to a cluster index associated with an utterance. However, since the index of one cluster can be interchanged with that of another cluster, multiple chromosomes may amount to an identical clustering result. For example, the chromosomes $\{1 1 1 2 2 3 3\}, \{1 1 1 3 3 2 2\}, \{2 2 2 1 1 3 3\}$, and $\{1 1 1 5 5 4 4\}$ represent the same clustering result derived by grouping seven utterances into three clusters. Such a non-unique representation of the solution would significantly increase the GA search space, and may lead to an inferior clustering result. To avoid this problem, we limit the inventory of
chromosomes to conform to a baseform representation defined as follows.

Let \( I(c_m) \) be the lowest index of the utterance in cluster \( c_m \).

Then, a chromosome is a baseform

\[
\text{iff } \forall c_m, c_l \neq \{\phi\}, \text{if } m < l, \text{then } I(c_m) < I(c_l),
\]

(8)

where \( \{\phi\} \) indicates that a cluster does not contain any utterance. Among the above chromosomes, \( \{1 1 1 2 2 3 3\} \) is a baseform, since the lowest index of the utterance in clusters \( c_1, c_2, \) and \( c_3 \) is 1, 4, and 6, respectively, which satisfies Eq. (8). In contrast, chromosomes \( \{1 1 1 3 3 2 2\} \) and \( \{2 2 2 1 1 3 3\} \) are not baseforms, since the lowest index of the utterance in clusters \( c_1, c_2, \) and \( c_3 \) does not satisfy Eq. (8). In addition, chromosome \( \{1 1 1 5 5 4 4\} \) implies that clusters \( c_2 \) and \( c_3 \) do not contain any utterance; hence it is not a baseform, either. However, it is conceivable that all the non-baseform chromosomes can be converted into a unique baseform representation by re-arranging the cluster indices.

GA optimization starts with a random generation of baseform chromosomes according to a certain population size, \( Z \). For example, for seven utterances to be clustered, we can generate chromosomes like \( \{1 1 1 2 2 3 3\}, \{1 1 2 3 3 4 2\}, \{1 2 2 2 2 2 1\}, \) and so on, in which the number of generated clusters is 3, 4, and 2, respectively. Then, the fitness of each chromosome is evaluated using the BIC-based score in Eq. (6). After this evaluation, a particular group of chromosomes is selected from the population to generate offspring by subsequent recombination. To prevent premature convergence of the population, the selection is performed with the linear ranking scheme [16], which sorts chromosomes in increasing order of fitness, and then assigns the expected number of offspring according to their relative ranking. Chromosomes with large fitness values will produce several copies, while chromosomes with tiny fitness values may be eliminated; hence, the total chromosome population size does not change.

Next, crossover among the selected chromosomes is performed by exchanging the substrings of two chromosomes between two randomly selected crossover points. A crossover probability is assigned to control the number of offspring produced in each generation. After crossover, a mutation operator is used to introduce random variations into the genetic structure of the chromosomes. This is done by generating a random number and then replacing one gene of an existing chromosome with a mutation probability. The resulting chromosomes that do not conform to the baseform representations are converted into their baseform counterparts. Then, the procedure of fitness evaluation, selection, crossover, and mutation is repeated continuously, in the hope that the overall fitness of the population will increase from generation to generation. When the maximum number of generations is reached, the best chromosome in the final population is taken as the solution, \( \textbf{H}^* \).

3. Experiments

3.1 Speech data

The speech data used in this study consisted of six excerpts of broadcasts from the evaluation set of the 2002 Rich Transcription Broadcast News and Conversational Telephone Speech Corpus [17]. Using the annotation files attached in the corpus, we segmented each excerpt into isolated utterances, each of which contains only one speaker’s voice. In our experiments, excerpt 1 was used as a development set to determine the appropriate values of parameters in the clustering methods, specifically, the number of mixture components used in the utterance-dependent GMM \( \lambda \), the chromosome population size, the crossover probability, and the mutation probability in GA optimization. Then, system performance was evaluated using excerpts 2, 3, 4, 5, and 6. Experiments were performed for each excerpt separately. Prior to the experiments, every utterance was converted from its digital waveform representation into a sequence of feature vectors. Each feature vector consisted of 12 Mel-scale frequency cepstral coefficients (MFCCs) and 12 delta MFCCs, computed using 20-ms Hamming window with 10-ms shift.

3.2 Performance evaluation metric

The performance of speaker clustering was evaluated on the basis of the Rand index [18], which represents the level of mis-clustering. Specifically, the Rand index (RI) is defined by the percentage of the number that two randomly-selected utterances from the same speaker are placed in different clusters, or that two randomly-selected utterances placed in the same cluster are from different speakers:

\[
R(M) = \frac{\sum_{i=1}^{M} n_{m}^{2} + \sum_{i=1}^{M} n_{p}^{2} - 2 \sum_{i=1}^{M} \sum_{j=1}^{M} n_{m}^{2} n_{p}^{2}}{\sum_{i=1}^{M} n_{m}^{2} + \sum_{i=1}^{M} n_{p}^{2} \times 100\%},
\]

(9)

where \( n_{m} \) is the number of utterances in cluster \( c_m \), \( n_{p} \) is the number of utterances produced by the \( p \)-th speaker, \( n_{mp} \) is the number of utterances in cluster \( c_m \) that were produced by the \( p \)-th speaker, and \( P \) is the total number of speakers. Obviously, the smaller the value of \( R(M) \), the better the clustering performance is.

3.3 Baseline systems

For the performance comparison, we also implemented two baseline systems, denoted as Baseline-I and Baseline-II, respectively. Baseline-I follows the conventional HAC framework [5,6]. It begins with each utterance as a single cluster and then successively merges the most similar pair of clusters. The similarities between clusters are computed using the complete linkage of the Generalized Likelihood Ratio (GLR)-based inter-utterance similarities. After a tree of clusters is generated, BIC presented in [6] was used to determine the optimal number of clusters. The second system, Baseline-II, stems from [9], which could be considered as a two-stage version of our proposed system. It begins by specifying a certain number of clusters, corresponding to one of the possible speaker population sizes, and then maximizes the level of overall within-cluster homogeneity using Eq. (2). The clustering method then examines various legitimate numbers of clusters by using Eq. (7) to determine the most likely speaker population size. Note that Eq. (7) in Baseline-II is used only to determine the most likely speaker population size, rather than optimizing the overall clustering as the proposed MBICC does.

3.4 Experimental results

We began the experiments by specifying the number of clusters \( a \text{ priori} \) as the true number of speakers. This could serve as an upper bound of the performance that the automatic determination of the speaker population size can achieve. Note that in this case, the proposed MBICC is equivalent to Baseline-II. Table 1 shows the speaker-clustering results. Here, the number of mixture components used in the utterance-dependent GMM \( \lambda \) was determined to be 64. The parameter values used for the maximum number of generations, the chromosome population size, the
crossover probability, and the mutation probability in GA optimization were determined to be 2000, 5000, 0.5, and 0.1, respectively. In addition, the penalty factor in BIC was set to one throughout all the systems. We can see from Table 1 that MBICC consistently yielded smaller values of the Rand index, compared with Baseline-I. This shows the superiority of global optimization applied in MBICC over pairwise optimization used in HAC.

We then examined the speaker-clustering performance of each system under the practical condition that the true speaker population size is unknown and must be estimated. Table 2 shows the clustering results. We can see from Table 2 that the number of speakers estimated by MBICC for each excerpt was very close to the true speaker population size. It is also clear that, for the estimated speaker population sizes, MBICC consistently yielded smaller values of the Rand index, compared with the two baseline systems. In addition, we can see that though Baseline-II has been shown superior to Baseline-I when the number of clusters is specified as the true speaker population, a large proportion of the Rand Index obtained with Baseline-II are larger than those of Baseline-I when the optimal number of clusters is determined automatically. This mainly because Baseline-II is prone to unreliable estimation of the optimal number of clusters. In contrast, MBICC is designed to jointly optimize the generated clusters and the number of clusters. The results validate the capability of MBICC to overcome the shortcomings of Baseline-I and Baseline-II.

Table 1. Speaker-clustering results obtained under a condition that the number of clusters is specified a priori as the true number of speakers.

<table>
<thead>
<tr>
<th>Excerpt</th>
<th># Ut.</th>
<th>True # Spk.</th>
<th>Baseline-I</th>
<th>MBICC (equivalent to Baseline-II)</th>
</tr>
</thead>
<tbody>
<tr>
<td>bn02en 2</td>
<td>29</td>
<td>9.6</td>
<td>5.2</td>
<td></td>
</tr>
<tr>
<td>bn02en 3</td>
<td>13</td>
<td>6.0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>bn02en 4</td>
<td>43</td>
<td>16.9</td>
<td>13.3</td>
<td></td>
</tr>
<tr>
<td>bn02en 5</td>
<td>26</td>
<td>10.7</td>
<td>16.2</td>
<td></td>
</tr>
<tr>
<td>bn02en 6</td>
<td>45</td>
<td>12.4</td>
<td>7.3</td>
<td></td>
</tr>
</tbody>
</table>

Table 2. Speaker-clustering results obtained with automatic determination of speaker population size.

<table>
<thead>
<tr>
<th>Excerpt</th>
<th>True # Spk.</th>
<th>Est. # Spk.</th>
<th>RI (%)</th>
<th>Est. # Spk.</th>
<th>RI (%)</th>
<th>Est. # Spk.</th>
<th>RI (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>bn02en 2</td>
<td>9</td>
<td>12.0</td>
<td>18.5</td>
<td>16</td>
<td>20.1</td>
<td>11</td>
<td>9.8</td>
</tr>
<tr>
<td>bn02en 3</td>
<td>6</td>
<td>6.0</td>
<td>0.0</td>
<td>6</td>
<td>0.0</td>
<td>6</td>
<td>0.0</td>
</tr>
<tr>
<td>bn02en 4</td>
<td>16</td>
<td>18.0</td>
<td>29.9</td>
<td>14</td>
<td>24.2</td>
<td>18</td>
<td>14.8</td>
</tr>
<tr>
<td>bn02en 5</td>
<td>10</td>
<td>11.3</td>
<td>13.1</td>
<td>14</td>
<td>33.1</td>
<td>11</td>
<td>16.9</td>
</tr>
<tr>
<td>bn02en 6</td>
<td>14</td>
<td>12.6</td>
<td>16.7</td>
<td>17</td>
<td>18.3</td>
<td>15</td>
<td>8.8</td>
</tr>
</tbody>
</table>

4. Conclusions

We have investigated techniques for clustering speech data, whereby utterances from the same speaker can be grouped into a single cluster. This requirement is formulated as a problem of jointly optimizing the homogeneity and the number of generated clusters, characterized by the Bayesian information criterion. Unlike the existing methods, in which BIC is used only to determine the optimal number of clusters, the proposed method uses BIC in conjunction with a genetic algorithm to determine the optimal cluster where each utterance should be located. As a result, we have demonstrated a noticeable improvement in the speaker-clustering performance, compared to the conventional method based on hierarchical agglomerative clustering and the Bayesian information criterion for the estimation of the speaker population size.

5. Acknowledgements

This work was supported in part by the National Science Council, Taiwan, under Grant NSC95-2218-E-027-020.

6. References