

Short Paper

Speaker Clustering Based on Bayesian Information Criterion *

WEI-HO TSAI

*Department of Electronic Engineering
Graduate Institute of Computer and Communication Engineering
National Taipei University of Technology
Taipei, 106 Taiwan*

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 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 at. The experimental results show that the proposed speaker-clustering method outperforms the conventional methods.

Keywords: speaker clustering, Bayesian information criterion, genetic algorithm, Gaussian mixture modeling, divergency

1. INTRODUCTION

With the burgeoning availability of digital audio material, it has become increasingly important to develop automated techniques for indexing or archiving the voluminous spoken data accumulated daily [1]. In addition to transcribing the linguistic meanings behind speech, determining “who spoken when” [2] in an audio stream is also a research topic of great interest in the recent years. This research topic generally involves two problems: 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 [5-17]. The two problems are often addressed jointly and termed speaker diarization [19-23]. It is hoped that, by locating utterances from the same speaker, the human effort required for indexing data with speaker identities can be greatly reduced from having to listen to every long audio recording to only having to check a few utterances in each cluster. In this paper, we concentrate on the latter problem, referred to as speaker clustering hereafter.

Received February 26, 2007; revised August 10, 2007; accepted October 24, 2007.

Communicated by Chin-Teng Lin.

* This paper was supported in part by the National Science Council of Taiwan, R.O.C. under grant No. NSC 95-2218-E-027-020. Part of this paper has been presented in the European Conference on Speech Communication and Technology, Aug 27-31, 2007, Antwerp, Belgium, ISCA.

In general, a speaker clustering system should work with no prior information regarding speakers involved and speaker population size. The task in question is thus defined as: given N speech utterances, each of which is assumed from one of the P unknown speakers, where $N \geq P$, how to partition the N utterances into M clusters, such that $M = P$, and each cluster consists of utterances from only one speaker? Currently, the most prevalent method for speaker clustering is a hierarchical clustering (HC) framework [4-12]. It computes the similarities of vocal characteristics between utterances, and then sequentially merges the utterances deemed similar to each other (agglomerative clustering), or alternatively, separates the utterances deemed dissimilar to each other (divisive clustering). The result of HC is a cluster tree, which contains N different partitionings with the number of clusters ranging from N to 1. To obtain the best clustering, the tree is then cut via an estimation of the speaker population size. Representative methods for estimating the speaker population size are based on the BBN Metric [5] and the Bayesian Information Criterion (BIC) [7].

Though the HC framework is popular for speaker clustering, it is far from optimal in a number of respects. First, the principle behind HC is to make the voice characteristics within a newly generated cluster as homogeneous as possible. However, it cannot 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. In particular, some mis-clustering errors, arising from grouping different-speaker utterances together, or separating same-speaker utterances, can propagate down the whole process, and hence limit the clustering performance. Second, the cluster tree is generated separately from the determination of the optimal number of clusters. Since 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, [16] proposes a clustering method, called Maximum Purity Clustering (MPC), to maximize the with-cluster homogeneity of speaker voice characteristics in a global fashion, instead of in a cluster-by-cluster manner used in HC. Meanwhile, [17] proposes a clustering method, called Minimum Divergency Clustering (MDC), which further integrates the similarity computation and cluster generation components into a unified optimization framework, so that the homogeneity for all the clusters can be summed to approach a maximum. However, both [16] and [17] follow the principle of conventional BIC-based method for estimating the speaker population size, which is performed separately from the generation of clusters; hence, they may still suffer from the problem of error propagation. On the other hand, [18] proposes a method called Minimum Rand Index Clustering (MRIC) to jointly optimize the generated clusters and the required number of clusters. However, though MRIC avoids the errors of mis-clustering from propagating to the estimation of speaker population size, it does not integrate the similarity computation and cluster generation components into a unified optimization framework like MDC.

To overcome the limitations of the above-mentioned methods, this study proposes jointly optimizing the generated clusters and the required number of clusters by maximizing a BIC value of clustering. As shown in Fig. 1, in which various clustering methods are summarized, the proposed method attempts to improve the clustering in two respects. First, in contrast to the HC framework, or MPC and MDC, in which BIC is used only to

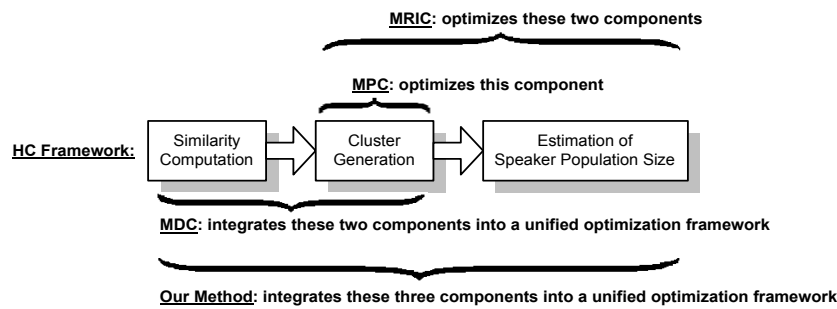


Fig. 1. A summarization of the HC framework, MPC, MDC, MRIC, and the proposed method.

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 at. Second, in contrast to MRIC, which performs the similarity computation and cluster generation as two separate processes, the proposed method combines the two processes into a unified optimization framework like MDC. The above concepts are carried out 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 the methods for characterizing the cluster homogeneity, representing the BIC-based scores, and maximizing the BIC-based scores.

The remainder of this paper is organized as follows. Section 2 introduces our proposed speaker clustering method, called maximum BIC clustering (MBICC). Section 3 presents our experimental results, along with a comparison to other speaker-clustering methods. In section 4, we conclude this paper with a summary.

2. MAXIMUM BIC CLUSTERING

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 converted into a problem of optimizing the score.

2.1 Scoring the Partitionings via the Bayesian Information Criterion

Our basic strategy for scoring a partitioning of the utterances is based on the Bayesian information criterion (BIC) [24]. The BIC is a model selection criterion, which evalu-

ates 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, $\Lambda_1, \Lambda_2, \dots, \Lambda_K$, built for characterizing a data set \mathbf{O} , then each model is scored via a BIC value:

$$\text{BIC}(\Lambda_k) = \log \Pr(\mathbf{O} | \Lambda_k) - 0.5\gamma\#(\Lambda_k) \log |\mathbf{O}|, \quad (1)$$

where $\Pr(\mathbf{O} | \Lambda_k)$ is the likelihood probability represents how Λ_k fits \mathbf{O} , $\#(\Lambda_k)$ denotes the number of free parameters in model Λ_k , $|\mathbf{O}|$ is the size of \mathbf{O} , and γ is a penalty factor. The second term in the right side of Eq. (1) represents an amount of penalty, thereby penalizing the model having high complexity, for the reason that a complicated model is expected to produce a larger value of likelihood probability than a simpler model does. The criterion favors the model having the largest value of BIC. Hence, 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 could be chosen as 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 [17], we characterize each possible partitioning as a set of indices $\mathbf{H} = \{h_1, h_2, \dots, h_N\}$ 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 at. Note that each index h_n is an integer between 1 and M , and the value of M is also unknown and to be determined. Let $\mathbf{X} = \{\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_N\}$ denote N utterances to be clustered, each of which is represented by a frame-based spectral feature stream, *i.e.*, $\mathbf{X}_n = \{\mathbf{x}_{n,1}, \mathbf{x}_{n,2}, \dots, \mathbf{x}_{n,T_n}\}$, with T_n frames, $1 \leq n \leq N$. Then, a partitioning is deemed homogeneous if $h_n = h_k$ for any utterances \mathbf{X}_n and \mathbf{X}_k similar to each other, and $h_n \neq h_k$ for any \mathbf{X}_n and \mathbf{X}_k dissimilar to each other.

To quantify the speaker homogeneity as a comparable value, we employ Gaussian mixture modeling technique [25], 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 $\lambda = \{w_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i, 1 \leq i \leq J\}$, where J is the number of mixed Gaussian densities, w_i are mixture weights, $\boldsymbol{\mu}_i$ are mean vectors, and $\boldsymbol{\Sigma}_i$ are covariance matrices. These parameters are estimated using the Expectation-Maximization (EM) algorithm [26]. After that, an utterance-dependent GMM $\lambda_n = \{w_{n,i}, \boldsymbol{\mu}_{n,i}, \boldsymbol{\Sigma}_{n,i}, 1 \leq i \leq J\}$ is formed for each utterance \mathbf{X}_n , $1 \leq n \leq N$, which represents a specific speaker's voice characteristics observed in \mathbf{X}_n . The parameters of λ_n are estimated by adapting λ based on Maximum A Posteriori (MAP) [27]. Next, for each cluster c_m , $1 \leq m \leq M$, we generate a cluster-dependent GMM $\lambda^{(m)} = \{w_j^{(m)}, \boldsymbol{\mu}_j^{(m)}, \boldsymbol{\Sigma}_j^{(m)}, 1 \leq j \leq J\}$ using all the utterances assigned to c_m . The parameters of $\lambda^{(m)}$ are also estimated by adapting λ 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 *divergency* [28], we can compute the speaker homogeneity for a partitioning \mathbf{H} using

$$\log \Pr(\mathbf{X} | \mathbf{H}) = \sum_{m=1}^M \sum_{n=1}^N \log \mathcal{S}(\lambda^{(m)}, \lambda_n) \delta(h_n, m), \quad (2)$$

where $\mathcal{S}(\lambda^{(m)}, \lambda_n)$ denotes the divergency-based model similarity [29]:

$$\mathcal{S}(\lambda^{(m)}, \lambda_n) = \sum_{j=1}^J \sum_{i=1}^J w_j^{(m)} w_{n,i} \exp[-\mathcal{D}(\boldsymbol{\mu}_j^{(m)}, \boldsymbol{\Sigma}_j^{(m)}; \boldsymbol{\mu}_{n,i}, \boldsymbol{\Sigma}_{n,i})], \quad (3)$$

$$\begin{aligned} \text{and } \mathcal{D}(\boldsymbol{\mu}_j^{(m)}, \boldsymbol{\Sigma}_j^{(m)}; \boldsymbol{\mu}_{n,i}, \boldsymbol{\Sigma}_{n,i}) &= \frac{1}{2} (\boldsymbol{\mu}_j^{(m)} - \boldsymbol{\mu}_{n,i})' (\boldsymbol{\Sigma}_j^{(m)-1} + \boldsymbol{\Sigma}_{n,i}^{-1}) (\boldsymbol{\mu}_j^{(m)} - \boldsymbol{\mu}_{n,i}) \\ &+ \frac{1}{2} \text{Tr}\{(\boldsymbol{\Sigma}_j^{(m)1/2} \boldsymbol{\Sigma}_{n,i}^{-1/2}) (\boldsymbol{\Sigma}_j^{(m)1/2} \boldsymbol{\Sigma}_{n,i}^{-1/2})'\} + \frac{1}{2} \text{Tr}\{(\boldsymbol{\Sigma}_j^{(m)-1/2} \boldsymbol{\Sigma}_{n,i}^{1/2}) (\boldsymbol{\Sigma}_j^{(m)-1/2} \boldsymbol{\Sigma}_{n,i}^{1/2})'\} - R, \end{aligned} \quad (4)$$

is the divergence between Gaussian distributions $\mathcal{N}(\boldsymbol{\mu}_j^{(m)}, \boldsymbol{\Sigma}_j^{(m)})$ and $\mathcal{N}(\boldsymbol{\mu}_{n,i}, \boldsymbol{\Sigma}_{n,i})$, $\text{Tr}(\cdot)$ 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.*, $w_j^{(m)} = w_{n,j} = w_j$, $1 \leq j \leq J$. Since the mixture components of $\lambda^{(m)}$ and λ_n are aligned, Eq. (3) can be simplified as

$$\mathcal{S}(\lambda^{(m)}, \lambda_n) = \sum_{j=1}^J w_j \exp[-\mathcal{D}(\boldsymbol{\mu}_j^{(m)}, \boldsymbol{\Sigma}_j^{(m)}; \boldsymbol{\mu}_{n,j}, \boldsymbol{\Sigma}_{n,j})]. \quad (5)$$

Consider the BIC in Eq. (1) for our clustering framework. The data set \mathbf{O} is the N utterance-dependent GMMs, which is further “modeled” by M cluster-dependent GMMs according to \mathbf{H} . Hence, the size of data are the number of utterance-dependent GMMs, *i.e.*, $|\mathbf{O}| = N$, 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 Λ 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 \mathbf{H} can be computed using

$$\mathcal{B}(\mathbf{H}) \equiv \sum_{m=1}^M \sum_{n=1}^N \log \mathcal{S}(\lambda^{(m)}, \lambda_n) \delta(h_n, m) - 0.5\gamma M \log N. \quad (6)$$

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

$$\mathbf{H}^* = \underset{\mathbf{H}}{\text{argmax}} \mathcal{B}(\mathbf{H}). \quad (7)$$

2.3 Optimizing the BIC-based Score via the Genetic Algorithm

Since the optimal number of clusters to be generated, M , is an unknown value be-

tween 1 and N , there are N^N possible solutions of \mathbf{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) [30] 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 \mathbf{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

$$\text{iff } \forall c_m, c_l \neq \{\phi\}, \text{ if } m < l, \text{ then } I(c_m) < I(c_l), \quad (8)$$

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.

Fig. 2 shows the flow diagram of GA optimization. It starts with a random generation of baseform chromosomes according to a certain population size, say Z . For example, for seven utterances to be clustered, we can generate chromosomes like $\{1\ 1\ 1\ 2\ 2\ 3\ 3\}$, $\{1\ 2\ 2\ 3\ 3\ 4\ 2\}$, $\{1\ 2\ 2\ 2\ 2\ 1\}$, 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 [31], 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

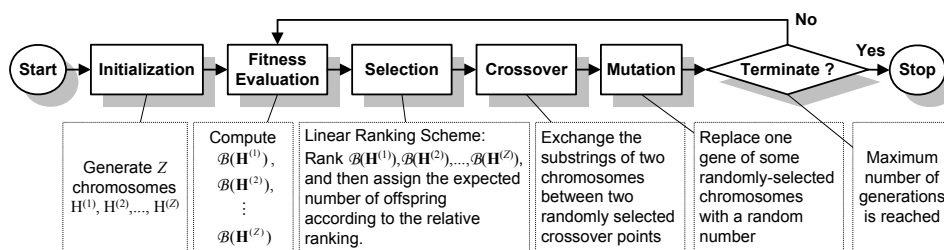


Fig. 2. Flow diagram of GA optimization.

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, say Q , is reached, the best chromosome in the final population is taken as the solution, H^* .

3. EXPERIMENTS

3.1 Speech Data

We used three data sets to conduct the speaker-clustering experiments. The first data set consisted of six excerpts of broadcasts from the evaluation set of the *2002 Rich Transcription Broadcast News and Conversational Telephone Speech Corpus* [32], denoted by RT-02. The speech of RT-02 was English, and it was digitized with a 16 kHz sampling rate and 16-bit quantization resolution. Using the annotation files attached in the corpus, we segmented each excerpt into isolated speech utterances, each of which contains only one speaker's voice. Table 1 summarizes the utterance duration and speaker population sizes in each excerpt of RT-02. The second data set contained 200 cellular telephone speech utterances extracted from the *2001 NIST Speaker Recognition Evaluation Corpus* [32], denoted by SRE-01. The speech of SRE-01 was English, and it was recorded with an 8 kHz sampling rate and 16-bit quantization resolution. The 200 utterances were produced by 20 randomly-selected speakers. The number of utterances spoken by each speaker was ten. The third data set contained 200 microphone-quality speech utterances extracted from TCC-300 [33]. Speech of TCC-300 was Mandarin, and it was recorded with a 16 kHz sampling rate and 16-bit quantization resolution. Similar to SRE-01, the 200 utterances of TCC-300 were produced by 20 randomly-selected speakers. The number of utterances spoken by each speaker was ten.

In our experiments, excerpt 1 of RT-02 was used as a development set to determine the appropriate values of parameters in the clustering methods, such as the number of mixture components used in the utterance-dependent GMM λ , the chromosome population size, the crossover probability, and the mutation probability in GA optimization.

Table 1. Speech data profile of RT-02.

Excerpt	# Utterances	Maximum/Minimum/Average Utterance Duration (in sec)	# Speakers
bn02en_1	44	67.8 / 0.5 / 13.0	16
bn02en_2	29	60.0 / 0.6 / 20.2	9
bn02en_3	13	84.8 / 5.0 / 37.8	6
bn02en_4	43	70.3 / 0.2 / 13.3	16
bn02en_5	26	65.0 / 5.0 / 23.1	10
bn02en_6	45	51.0 / 1.8 / 12.7	14

Then, system performance was evaluated using excerpts 2, 3, 4, 5, and 6, in which the experiments were performed for each excerpt separately. In addition, we further evaluated the system performance using SRE-01 and TCC-300, which represent different speech quality and spoken language from RT-02. 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 (frame) with 10-ms frame shift.

3.2 Performance Evaluation Metrics

The performance of speaker clustering was evaluated on the basis of the Rand index [5, 34], which measures the level of mis-clustering. Specifically, the Rand index is defined by the percentage¹ 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:

$$R(M) = \frac{\sum_{m=1}^M n_{m^*}^2 + \sum_{p=1}^P n_{*p}^2 - 2 \sum_{m=1}^M \sum_{p=1}^P n_{mp}^2}{\sum_{m=1}^M n_{m^*}^2 + \sum_{p=1}^P n_{*p}^2} \times 100\%, \quad (9)$$

where n_{m^*} is the total number of utterances in cluster c_m , n_{*p} is the total 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. Perfect clustering should produce a Rand Index of zero. In general, the Rand index decreases with an increase in the value of M initially, and reaches the minimum around $M = P$. When $M \gg P$, the Rand index increase as the value of M increases.

In addition, recognizing the fact that in many applications, assigning a long utterance into a wrong cluster can be more detrimental than assigning a short utterance into a wrong cluster, we also compute the Rand Index in a frame level, rather than only in the above-mentioned utterance level. Specifically, a frame-based Rand Index is defined by the percentage of the number that two randomly-selected frames from the same speaker

¹ We use the “percentage” instead of the “number” originally defined in [5, 34], because its value is more easily perceivable.

are placed in different clusters, or that two randomly-selected frames placed in the same cluster are from different speakers. Thus, in using Eq. (9) to compute a frame-based Rand Index, n_m^* represents the total number of frames in cluster c_m , n_{*p} is the total number of frames produced by the p th speaker and n_{mp} represents the number of frames in cluster c_m produced by the p th speaker.

3.3 The Baseline Systems

For performance comparison, we also implemented two baseline systems, denoted as Baseline-I and Baseline-II, respectively. The first system, Baseline-I, follows the hierarchical agglomerative clustering framework, which is the most popular speaker-clustering approach [7]. 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. Briefly, the GLR is defined as

$$S(\mathbf{X}_i, \mathbf{X}_j) = \frac{\Pr(\mathbf{X}_{ij} | \lambda_{ij})}{\Pr(\mathbf{X}_i | \lambda_i) \Pr(\mathbf{X}_j | \lambda_j)}, \quad (10)$$

where \mathbf{X}_{ij} is the concatenation of two utterances \mathbf{X}_i and \mathbf{X}_j , and λ_i , λ_j , and λ_{ij} are GMMs, trained using \mathbf{X}_i , \mathbf{X}_j , and \mathbf{X}_{ij} , respectively. To avoid the situation that some utterance is too short to train a GMM reliably, all the GMMs are created by adapting the utterance-dependent GMM λ . After the cluster tree is generated, BIC presented in [7] was used to determine the optimal number of clusters.

The second system, Baseline-II, stems from [17], 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.

Before conducting experiments, the computational complexities of the above three systems are compared. We observe that there are two factors which dominate the overall computational time for the three systems. The first factor arises from the Gaussian mixture modeling of feature vectors, *e.g.*, the generation of λ_{ij} in Eq. (10), or the generation of cluster model $\lambda^{(m)}$ in Eq. (2). The second factor arises from the computation of Gaussian functions based on the models, *e.g.*, $\Pr(\mathbf{X}_{ij} | \lambda_{ij})$, or $\mathcal{S}(\lambda^{(m)}, \lambda_n)$. However, since the models are generated using the MAP adaptation, the first factor can be ignored, compared to the second factor. The system complexity depends mainly on how many Gaussian functions need to be performed. As analyzed in [6], the computational complexities of Baseline-I and Baseline-II can be characterized by $O(N^2JT/2)$ and $O(N^2JZQ)$, respectively, in which T is the total number of feature vectors in the utterance collection, Z is the number of chromosomes, and Q is the number of generations in GA. Because MBICC and Baseline-II perform the same amount of computation for Gaussian functions, their complexities are similar, which are around $2ZQ/T$ times the complexity of Baseline-I.

3.4 Experimental Results

We began the experiments by determining the tunable parameters involved in each system using excerpt 1 of RT-02. In Baseline-I, the number of mixture components used in the utterance-dependent GMM λ was determined to be 32. In Baseline-II and MBICC, the number of mixture components used in the utterance-dependent GMM λ 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.

Tables 2 (a), (b), and (c) show the speaker-clustering results of RT-02, obtained with Baseline-I, Baseline-II, and the proposed MBICC, respectively. In the left portion of

Table 2. Speaker-clustering results for data set RT-02.

(a) Baseline I.

Excerpt	# Clusters = True # Speakers			# Clusters = Estimated # Speakers		
	True # Speakers	Utterance-based Rand Index (%)	Frame-based Rand Index (%)	Estimated # Speakers	Utterance-based Rand Index (%)	Frame-based Rand Index (%)
bn02en_2	9	6.6	8.0	12	18.5	19.9
bn02en_3	6	0.0	0.0	6	0.0	0.0
bn02en_4	16	20.9	18.6	18	29.9	26.4
bn02en_5	10	27.7	22.7	11	31.3	27.8
bn02en_6	14	12.4	21.1	12	16.7	22.4

(b) Baseline II.

Excerpt	# Clusters = True # Speakers			# Clusters = Estimated # Speakers		
	True # Speakers	Utterance-based Rand Index (%)	Frame-based Rand Index (%)	Estimated # Speakers	Utterance-based Rand Index (%)	Frame-based Rand Index (%)
bn02en_2	9	5.2	6.9	16	20.1	22.6
bn02en_3	6	0.0	0.0	6	0.0	0.0
bn02en_4	16	13.3	11.4	14	24.2	19.6
bn02en_5	10	16.2	10.1	14	33.1	26.9
bn02en_6	14	7.3	10.8	17	18.3	19.9

(c) MBICC.

Excerpt	# Clusters = True # Speakers			# Clusters = Estimated # Speakers		
	True # Speakers	Utterance-based Rand Index (%)	Frame-based Rand Index (%)	Estimated # Speakers	Utterance-based Rand Index (%)	Frame-based Rand Index (%)
bn02en_2	9	5.2	6.9	11	9.8	10.3
bn02en_3	6	0.0	0.0	6	0.0	0.0
bn02en_4	16	13.3	11.4	18	14.8	12.1
bn02en_5	10	16.2	10.1	11	16.9	11.3
bn02en_6	14	7.3	10.8	15	8.8	13.2

each table, we show the speaker-clustering performance under a condition that the number of clusters is specified a 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. We can see from Table 2 that MBICC or Baseline-II 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 HC.

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. It can be seen 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. By 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.

Next, experiments were conducted using SRE-01 and TCC-300 to further examine the performance of MBICC. The required numbers of clusters were determined automatically. Table 3 shows the results. We can see from Table 3 that regardless of speech quality and spoken language, MBICC consistently yielded smaller values of the Rand index, compared with the two baseline systems. The results confirm the validity of the proposed method.

Table 3. Speaker-clustering results for data sets SRE-01 and TCC-300.

(a) SRE-01 (true number of speakers is 20).

Method	Estimated # Speakers	Utterance-based Rand Index (%)	Frame-based Rand Index (%)
Baseline I	27	31.1	27.9
Baseline II	31	30.8	25.2
MBICC	25	24.2	22.3

(b) TCC-300 (true number of speakers is 20).

Method	Estimated # Speakers	Utterance-based Rand Index (%)	Frame-based Rand Index (%)
Baseline I	22	14.2	11.2
Baseline II	23	13.9	10.8
MBICC	22	12.5	9.7

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 formu-

lated 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 at. 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.

REFERENCES

1. J. Makhoul, F. Kubala, T. Leek, D. Liu, L. Nguyen, R. Schwartz, and A. Srivastava, "Speech and language technologies for audio indexing and retrieval," in *Proceedings of IEEE*, Vol. 88, 2000, pp. 1338-1353.
2. S. E. Johnson, "Who spoke when? – Automatic segmentation and clustering for determining speaker turns," in *Proceedings of European Conference on Speech Communication and Technology*, 1999, pp. 2211-2214.
3. B. Zhou and J. H. L. Hansen, "Unsupervised audio stream segmentation and clustering via the Bayesian information criterion," in *Proceedings of International Conference on Spoken Language Processing*, 2000, pp. 714-717.
4. M. A. Siegler, U. Jain, B. Raj, and R. M. Stern, "Automatic segmentation, classification and clustering of broadcast news audio," in *Proceedings of DARPA Speech Recognition Workshop*, 1997, pp. 97-99.
5. A. Solomonoff, A. Mielke, M. Schmidt, and H. Gish, "Clustering speakers by their voices," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1998, pp. 757-760.
6. D. A. Reynolds, E. Singer, B. A. Carson, G. C. O'Leary, J. J. McLaughlin, and M. A. Zissman, "Blind clustering of speech utterances based on speaker and language characteristics," in *Proceedings of International Conference on Spoken Language Processing*, 1998, pp. 3193-3196.
7. S. S. Chen and P. S. Gopalakrishnan, "Clustering via the Bayesian information criterion with applications in speech recognition," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1998, pp. 645-648.
8. R. Faltlhauser and G. Ruske, "Robust speaker clustering in eigenspace," in *Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding*, 2001, pp. 57-60.
9. H. Jin, F. Kubala, and R. Schwartz, "Automatic speaker clustering," in *Proceedings of DARPA Speech Recognition Workshop*, 1997, pp. 108-111.
10. D. Liu and F. Kubala, "Online speaker clustering," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2003, pp. 333-336.
11. Z. Liu, "An efficient algorithm for clustering short spoken utterances," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2005, pp. 593- 596.
12. Y. Moh, P. Nguyen, and J. C. Junqua, "Towards domain independent speaker clustering," in *Proceedings of IEEE International Conference on Acoustics, Speech, and*

- Signal Processing*, 2003, pp. 85-88.
13. S. E. Johnson and P. C. Woodland, "Speaker clustering using direct maximization of the MLLR-adapted likelihood," in *Proceedings of International Conference on Spoken Language Processing*, 1998, pp. 1775-1778.
 14. J. Ajmera, H. Bourlard, I. Lapidot, and I. McCowan, "Unknown-multiple speaker clustering using HMM," in *Proceedings of International Conference on Spoken Language Processing*, 2002, pp. 573-576.
 15. I. Lapidot, H. Guterman, and A. Cohen, "Unsupervised speaker recognition based on competition between self-organizing maps," *IEEE Transactions on Neural Network*, Vol. 13, 2002, pp. 877-887.
 16. W. H. Tsai and H. M. Wang, "Speaker clustering of unknown utterances based on maximum purity estimation," in *Proceedings of European Conference on Speech Communication and Technology*, 2005, pp. 3069-3072.
 17. W. H. Tsai and H. M. Wang, "Speech Utterance clustering based on the maximization of within-cluster homogeneity of speaker voice characteristics," *The Journal of the Acoustical Society of America*, Vol. 120, 2006, pp. 1631-1645.
 18. W. H. Tsai and H. M. Wang, "Speaker clustering based on minimum rand index," in *Proceedings of IEEE International Conference on Acoustics, Speech, Signal Processing*, 2007, pp. 306-309.
 19. NIST, "Benchmark tests: rich transcription," <http://www.nist.gov/speech/tests/rt/rt-2005/spring/index.htm>.
 20. M. Ben, M. Betser, F. Bimbot, and G. Gravier, "Speaker diarization using bottom-up clustering based on a parameter-derived distance between adapted GMMs," in *Proceedings of International Conference on Spoken Language Processing*, 2004, pp. 2329-2332.
 21. S. E. Tranter, "Two-way cluster voting to improve speaker diarisation performance," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2005, pp. 753-756.
 22. R. Sinha, S. E. Tranter, M. J. F. Gales, and P. C. Woodland, "The Cambridge University March 2005 speaker diarisation system," in *Proceedings of European Conference on Speech Communication and Technology*, 2005, pp. 2437-2440.
 23. X. Zhu, C. Barras, S. Meignier, and J. L. Gauvain, "Combining speaker identification and BIC for speaker diarization," in *Proceedings of European Conference on Speech Communication and Technology*, 2005, pp. 2441-2444.
 24. G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, Vol. 6, 1978, pp. 461-464.
 25. D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Transactions on Speech and Audio Processing*, Vol. 3, 1995, pp. 72-83.
 26. A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, Vol. 39, 1977, pp. 1-38.
 27. D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, Vol. 10, 2000, pp. 19-41.
 28. S. Kullback, *Information Theory and Statistics*, Dover, New York, 1968.
 29. C. S. Huang, H. C. Wang, and C. H. Lee, "A study on model-based error rate estima-

- tion for automatic speech recognition,” *IEEE Transactions on Speech and Audio Processing*, Vol. 11, 2003, pp. 581-589.
30. D. E. Goldberg, *Genetic Algorithm in Search, Optimization and Machine Learning*, Addison-Wesley, New York, 1989.
 31. J. E. Baker, “Adaptive selection methods for genetic algorithm,” in *Proceedings of International Conference on Genetic Algorithms and Their Applications*, 1985, pp. 101-111.
 32. LDC, <http://www ldc.upenn.edu/>.
 33. ACLCLP, http://www.aclclp.org.tw/corp_c.php.
 34. W. M. Rand, “Objective criteria for the evaluation of clustering methods,” *Journal of the American Statistical Association*, Vol. 66, 1971, pp. 846-850.

Wei-Ho Tsai (蔡偉和) received his B.S. degree in Electrical Engineering from National Sun Yat-Sen University, Kaohsiung, Taiwan, R.O.C., in 1995. He received his M.S. and Ph.D. degrees in Communication Engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1997 and 2001, respectively. From 2001 to 2003, he was with Philips Research East Asia, Taipei, Taiwan, where he worked on speech processing problems in embedded systems. From 2003 to 2005, he served as a Postdoctoral Fellow at the Institute of Information Science, Academia Sinica, Taipei, Taiwan. He is currently an Assistant Professor in the Department of Electronic Engineering & Graduate Institute of Computer and Communication Engineering, National Taipei University of Technology, Taiwan. His research interests include spoken language processing and music information retrieval.