ONLINE SPEAKER CLUSTERING USING INCREMENTAL LEARNING OF AN ERGODIC HIDDEN MARKOV MODEL

Takafumi KOSHINAKA[*], *Kentaro NAGATOMO*[*], *and Koichi SHINODA*[†]

[*] Common Platform Software Res. Labs., NEC Corporation, Kawasaki, Japan
[†] Dept. of Computer Science, Tokyo Institute of Technology, Tokyo, Japan

ABSTRACT

A novel online speaker clustering method suitable for real-time applications is proposed. Using an ergodic hidden Markov model, it employs incremental learning based on a variational Bayesian framework and provides probabilistic (non-deterministic) decisions for each input utterance, directly considering the specific history of preceding utterances. It makes possible more robust cluster estimation and precise classification of utterances than do conventional online methods. Experiments on meeting-speech data show that the proposed method produces 70–80% fewer errors than a conventional method does.

Index Terms— HMM, variational Bayesian algorithm, model selection, meeting recognition

1. INTRODUCTION

Speaker clustering is a technique that classifies speech utterances from multiple speakers as observed in broadcast news, meetings, etc. so that utterances from a given speaker will be assigned a unique label (cluster). Clustering output is commonly used in unsupervised speaker adaptation in recently developed large-vocabulary continuous speech recognition (LVCSR) systems to achieve higher recognition accuracy [1].

Most conventional clustering methods belong to a family of batch processing methods [2] which typically require that all utterances be present before clustering can be executed, and in this, they do not meet the requirements for real-time applications.

Online algorithms have also been proposed [3, 4]. The basic idea behind these algorithms is deterministic and successive classification according to a kind of (dis)similarity measurement between an input utterance and existing clusters.

Deterministic classification (hard decision), however, has difficulty in dealing with low confidence-level decisions, which easily arise when the duration of input utterances is too short, or when there exist two or more speakers quite similar in vocal characteristics to one another. Since any utterance will be classified into a unique cluster whether the confidence-level of the decision is high or not, utterances resulting in low confidence-levels often cause misclassification that will accumulate in the clusters and disturb cluster estimation. Further, clusters incorrectly estimated in this way may cause still more misclassification.

Further, the similarity measures used in existing online algorithms involve no more than calculating a similarity between an input utterance and currently existing clusters. Such measures are based on the unrealistic assumption that there existed no mistake before the current decision. We have believed that more precise classification might be attained by conducting cluster estimation correction that takes into account the possibility of misclassification of preceeding utterances.

In this paper, we propose a stochastic online speaker clustering method in which incremental learning [5] is applied to a generative model of speech utterances [6, 7]. It repeatedly estimates model parameters and a probabilistic clustering result for each input utterance, and updates those for a certain number of preceding utterances. The effectiveness of the proposed method is demonstrated through a series of experiments using meeting-speech data.

This paper is organized as follows: Section 2 describes existing batch approaches based on generative models, i.e., ergodic hidden Markov models (HMMs); in Section 3, we present a Bayesian formulation for our online algorithm, based on the generative model-based approach; Section 4 gives experimental results, and in Section 5 we summarize our work and discuss future issues.

2. MODEL-BASED CLUSTERING

Let us consider an N-state ergodic HMM (Fig. 1). This HMM starts operation from the initial state I and makes a transition to the state $i \in \{1, \cdots, N\}$ according to the transition probability π_i. While under the state i, it repeatedly outputs feature vectors $(\boldsymbol{x})$ in accord with a probability density function $f_i(\boldsymbol{x})$. Its operation is completed when the final state F is attained at the probability a_i. At this point, a sequence of feature vectors, $X = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_T)$ is generated.

Assuming a one-to-one correspondence between states and speakers, this HMM can be regarded as a generative model of a speech utterance from a speaker selected out of a

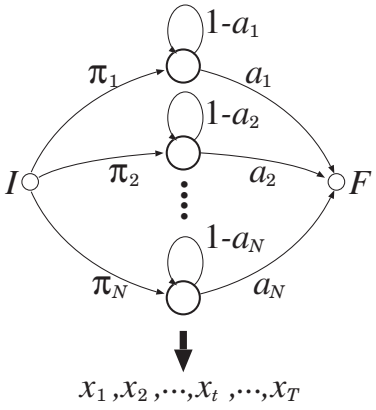

Fig. 1. Generative model of speech utterances from N speakers.

set of N speakers. Given a set of model parameters θ and an utterance $X = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_T)$, the probability that speaker i has given the utterance X can be calculated as follows:

$$P(i \mid X, \theta) \propto \pi_i a_i (1 - a_i)^{T-1} \prod_{t=1}^{T} f_i(\boldsymbol{x}_t). \quad (1)$$

Given n utterances $X_1, \cdots, X_n$, the speaker clustering procedure will be as follows: 1) estimate θ using n utterances, 2) calculate Eq. (1) using the resulting θ, and 3) select the i which maximizes $P(i \mid X, \theta)$.

Steps in the above procedure are ordinarily performed in succession, i.e., in batch processing. In the next section, we formulate an online version of the above speaker clustering procedure.

3. ONLINE ALGORITHM

3.1. Hyper-parameter reestimation

Online learning by stochastic models that include hidden variables can be accomplished on the basis of a generalized EM (GEM) algorithm [5], which employs a maximum negative free energy criterion. Here for i.i.d. (independent and identically distributed) observations (utterances) $X_1, X_2, \cdots$, E- and M-steps will be incrementally performed for each observation X_t. The difference between online (GEM-based) and batch (traditional EM) algorithms is the E-step. At the E-step in the online algorithm, the probability distribution for the hidden variables is updated only for the latest observation, rather than for all observations to that point.

Here, we employ a Bayesian approach rather than the standard maximum likelihood (ML) approach, and use a variational Bayesian (VB) algorithm [8] as a learning algorithm, as is done in [9]. The VB algorithm is performed in two-steps, i.e., Bayesian E- and M-steps, and an online learning algorithm can be derived in the same way as a GEM algorithm.

The following gives a simple description of a hyper-parameter reestimation formula for an HMM. Here, for simplicity, we assume that the probability distribution for feature vectors is a Gaussian, $f_i(\boldsymbol{x}) = f(\boldsymbol{x} \mid \boldsymbol{\mu}_i, \Sigma_i)$.

First, we assume a parametric prior distribution for the parameter set θ as follows:

$$p(\theta) = \mathcal{D}\left(\pi_1, \cdots, \pi_N \mid \lambda_1^{(0)}, \cdots, \lambda_N^{(0)}\right) \times \prod_{i=1}^{N} \mathcal{B}\left(a_i \mid \kappa_{0,i}^{(0)}, \kappa_{1,i}^{(0)}\right) \mathcal{N}\left(\boldsymbol{\mu}_i \mid \boldsymbol{\nu}_i^{(0)}, \xi_i^{(0)} \Sigma_i\right) \mathcal{W}\left(\Sigma_i \mid \eta_i^{(0)}, B_i^{(0)}\right), \quad (2)$$

where $\mathcal{D}$, $\mathcal{B}$, $\mathcal{N}$, and $\mathcal{W}$ denote, respectively, Dirichlet, beta, normal, and Wishart distributions. Eq. (2) is known as a conjugate prior, and here we refer to $\lambda_i^{(0)}$, $\kappa_{0,i}^{(0)}$, $\kappa_{0,i}^{(0)}$, $\boldsymbol{\nu}_i^{(0)}$, $\xi_i^{(0)}$, $\eta_i^{(0)}$, and $B_i^{(0)}$ as hyper-parameters for prior distributions.

The goal for the VB algorithm here is to obtain locally optimal hyper-parameters for posterior distributions: λ_i, $\kappa_{0,i}$, $\kappa_{0,i}$, $\boldsymbol{\nu}_i$, ξ_i, η_i, and B_i, for a given sequence of utterances and given hyper-parameters for a prior distribution. An online hyper-parameter reestimation formula and speaker clustering algorithm can then be derived as follows:

Initialization: Set hyper-parameters for the prior distribution $\lambda_i^{(0)}$, $\kappa_{0,i}^{(0)}$, $\kappa_{1,i}^{(0)}$, $\boldsymbol{\nu}_i^{(0)}$, $\xi_i^{(0)}$, $\eta_i^{(0)}$, $B_i^{(0)}$ to appropriately small, positive values. Set sufficient statistics variables $S_{00,i}$, $S_{0,i}$, $\boldsymbol{S}_{1,i}$, $S_{2,i}$ to zero.

E-step: Given the nth utterance $X_n = (\boldsymbol{x}_{n,1}, \cdots, \boldsymbol{x}_{n,T_n})$, calculate the Bayesian expectation $\bar{z}_{ni} = P(i \mid X_n)$ of hidden variables. This represents the probability that utterance X_n belongs to the ith cluster. Then, update the sufficient statistics variables as follows:

$$S_{00,i} \leftarrow S_{00,i} + \bar{z}_{ni}, \quad S_{0,i} \leftarrow S_{0,i} + T_n \bar{z}_{ni},$$

$$\boldsymbol{S}_{1,i} \leftarrow \boldsymbol{S}_{1,i} + \bar{z}_{ni} \sum_{t=1}^{T_n} \boldsymbol{x}_{nt},$$

$$S_{2,i} \leftarrow S_{2,i} + \bar{z}_{ni} \sum_{t=1}^{T_n} \boldsymbol{x}_{nt} \boldsymbol{x}_{nt}^T. \quad (3)$$

M-step: Update the hyper-parameters for the posterior distribution as follows:

$$\kappa_{0,i} \leftarrow \kappa_{0,i}^{(0)} + S_{0,i} - S_{00,i}, \quad \kappa_{1,i} \leftarrow \kappa_{1,i}^{(0)} + S_{00,i},$$

$$\lambda_i \leftarrow \lambda_i^{(0)} + S_{00,i}, \quad \boldsymbol{\nu}_i \leftarrow \frac{\xi_i^{(0)} \boldsymbol{\nu}_i^{(0)} + \boldsymbol{S}_{1,i}}{\xi_i^{(0)} + S_{0,i}},$$

$$\xi_i \leftarrow \xi_i^{(0)} + S_{0,i}, \quad \eta_i \leftarrow \eta_i^{(0)} + S_{0,i},$$

$$B_i \leftarrow B_i^{(0)} + S_{2,i} - 2\boldsymbol{S}_{1,i}\bar{\boldsymbol{x}}_i^T + S_{0,i}\bar{\boldsymbol{x}}_i\bar{\boldsymbol{x}}_i^T + \frac{\xi_i^{(0)} S_{0,i}}{\xi_i^{(0)} + S_{0,i}} \left(\bar{\boldsymbol{x}}_i - \boldsymbol{\nu}_i^{(0)}\right)\left(\bar{\boldsymbol{x}}_i - \boldsymbol{\nu}_i^{(0)}\right)^T, \quad (4)$$

where $\bar{\boldsymbol{x}}_i = \boldsymbol{S}_{1,i}/S_{0,i}$. Perform the E- and M-steps iteratively until convergence.

Note that expectation $\bar{z}_{ni}$ is calculated only for the latest data X_n, while expectations regarding the older data $\bar{z}_{1,i}, \cdots, \bar{z}_{n-1,i}$ are discarded after being added to the sufficient statistics variables in accord with Eq. (3).

The algorithm derived as just described can be extended to a "multi-buffer" version, in which the latest b utterances $X_{n-b+1}, \cdots, X_n$ ($b > 1$) are kept in a buffer area, and b expectations $\bar{z}_{n-b+1,i}, \cdots, \bar{z}_{n,i}$ are updated for each utterance input. In the experiments below, we investigate the effect of increasing buffer size b.

3.2. Model selection

The algorithm in the previous subsection has been described on the assumption that the number of speakers (clusters) N is known, but this is generally unknown in practical speaker clustering problems. In online cases, in particular, N is likely to change dynamically, starting from $N = 1$, and online estimation of the number of clusters is an essential problem.

With our proposed method, the hyper-parameter reestimation algorithm (3)–(4) is executed twice for each input utterance, first on the assumption that the number of clusters in the current observation is equal to that in the previous observation N, and next that it is equal to $N + 1$ (Fig. 2). Model selection is conducted to select either N or $N + 1$ on the basis of the following hypothesis test:

$$\log P\left(N + 1 \mid X_n\right) - \log P\left(N \mid X_n\right) > T, \quad (5)$$

where $P\left(N \mid X_n\right)$ is a Bayesian posterior probability with respect to the number of clusters. This probability can be calculated on the basis of the VB framework. That is, if Eq. (5) holds, then the hypothesis $N + 1$ is selected. T is a threshold value that has to be manually tuned beforehand on the basis of data used in its original development.

4. EXPERIMENTS

4.1. Experimental setup

We used a total of two hours of broadcast audio data (22 kHz, 16 bit PCM) from committee meetings of a Japanese governmental assembly and involving a total of 30 speakers. We applied simple VAD (voice activity detection) and acoustic analysis to the audio data and obtained 3,084 utterances (sequences of 12-dimentional MFCC feature vectors). The average duration of utterances was 1.90, and ranged from 0.24 to 10.86 seconds. We divided these utterances into six subsets,

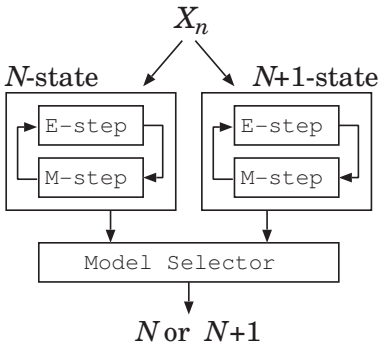

Fig. 2. Model selection for determining number of clusters

each of which contained 514 utterances given by 6–10 speakers for the purpose of performing the leave-one-out evaluation described below.

We employed an evaluation measure called the Rand Index [4], which is commonly used in such data partitioning problems. The Rand Index represents a kind of error rate and is defined as the probability that two randomly selected utterances assigned to different clusters are actually from the same speaker, or that they are from different speakers but have been assigned to the same cluster.

In our experimental setup, we used a conventional method called leader-follower clustering (LFC) [3], which is a generic approach to online clustering and deterministically classifies utterances according to an arbitrarily defined similarity measure. Here, we defined a similarity measure between each input utterance and existing clusters on the basis of the Bayesian information criterion (BIC) [1], where each cluster was assumed to be Gaussian. Since such a BIC-based measure contains a threshold value like the T in Eq. (5), LFC makes repeated determinations on the basis of the sign of the measure as to whether to merge the input utterance with the closest existing cluster or to create a new cluster.

4.2. Experimental results

First, we investigated the relationship between the Rand Index and the number of clusters by varying threshold value settings. We also tried three buffer sizes $b = 1, 2, 4$ for the proposed method. Results show that, for any number of clusters, the proposed method with the basic setting ($b = 1$) yields better performance than LFC does (see "Conventional" and "Proposed(1)" in Fig. 3). We also found that the proposed method shows still higher performance when buffer size b is increased to 2 or 4 (see "Proposed(2)" and "Proposed(4)"). Notably, when the number of clusters is comparable to the true number of speakers (# of cluster / # of speakers ≈ 1), our method with $b = 4$ produces 75% fewer errors than LFC does.

We also conducted experiments under the more practical condition that the threshold value T was fixed after being

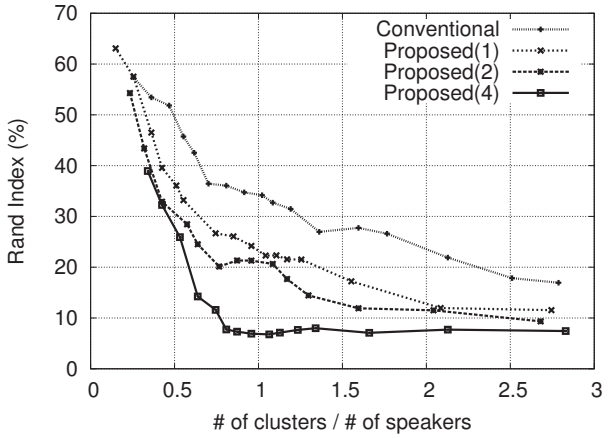

Fig. 3. Rand Index vs. number of clusters normalized with the true number of speakers.

tuned on the basis of data used in its development. To do this, we divided the six subsets of utterances into two groups. Specifically, we selected one subset for testing, and the rest were kept as development data for tuning the threshold value. All the subsets were rotated between test and development data (i.e., we performed a leave-one-out evaluation). Here we tuned the threshold value so that the total number of clusters would be as close as possible to that of the speakers.

Table 1. Leave-one-out evaluation for speaker clustering and estimation of the number of clusters.

	Av. Rand Index (%)	# of clusters/ # of speakers
Conventional	38.1	47 / 47
Proposed(1)	27.4	45 / 47
Proposed(2)	24.6	48 / 47
Proposed(4)	8.07	50 / 47
Batch	9.61	45 / 47

Table 1 shows results for average Rand Index and for estimation accuracy with respect to the number of clusters. It also shows results for the batch version of the proposed method. As may be seen, the proposed method outperforms the conventional method in clustering accuracy, and still better performance can be obtained with it by increasing buffer size b. We may also note that, somewhat unexpectedly, $b = 4$ with the proposed method yields performance that is comparable to or better than that with the batch approach, even though the latter has, from the very beginning of the clustering process, access to full information about the input utterances. Estimation of the number of clusters seems to work well enough, on average, with both the conventional and proposed methods.

5. SUMMARY AND FUTURE WORK

In this paper, we have proposed a novel online algorithm for speaker clustering that was formulated by applying a variational Bayesian incremental learning technique to a generative model of speech utterances from multiple speakers. The effectiveness of the proposed method has been demonstrated in a series of experiments using actual recorded-meeting audio data. Notably, the proposed method was found to be able to achieve still higher accuracy when buffer size b was increased.

Issues for the future include confirming the validity of the proposed method through its testing on evaluation measures other than the Rand Index, such as cluster/speaker purity. Also, further improvements in accuracy might be obtained by using prior knowledge, such as large-scale speaker databases. We would also like to investigate performance when our algorithm is implemented on LVCSR systems, where the accuracy of its speaker clustering might be expected to make the online speaker adaptation process work more effectively.

6. REFERENCES

[1] S. S. Chen *et al.*, "Automatic transcription of broadcast news," *Speech Communication*, vol. 37, pp. 69–87, 2002.

[2] T. Hain *et al.*, "Segment generation and clustering in the HTK broadcast news transcription system," *DARPA Broadcast News Transcription and Understanding Workshop*, pp. 133–137, 1998.

[3] R. Duda *et al.*, "Pattern classification (Second edition)," *John Wiley & Sons Inc.*, 2001.

[4] D. Liu and F. Kubala, "Online speaker clustering," *Proc. ICASSP2004*, vol. I, pp. 333–336, 2004.

[5] R. M. Neal and G. E. Hinton, "A view of the EM algorithm that justifies incremental, sparse, and other variants," *Learning in Graphical Models, The MIT Press*, pp. 355–368, 1998.

[6] J. Ajmera *et al.*, "Unknown-multiple speaker clustering using HMM," *Proc. ICSLP2002*, pp. 573–576, 2002.

[7] J. O. Olsen, "Separation of speakers in audio data," *Proc. EUROSPEECH95*, pp. 355–358, 1995.

[8] H. Attias, "Inferring parameters and structure of latent variable models by variational Bayes," *15th Conf. on Uncertainty in Artificial Intelligence*, pp. 21–30, 1999.

[9] F. Valente and C. Wellekens, "Variational Bayesian adaptation for speaker clustering," *Proc. ICASSP2005*, vol. I, pp. 965–968, 2005.