SOM as Likelihood Estimator for Speaker Clustering

Itshak Lapidot

Dalle Molle Institute for Perceptual Artificial Intelligence (IDIAP)
P.O.BOX 592, CH-1920 Martigny, Switzerland
lapidot@idiap.ch

Abstract

A new approach is presented for clustering the speakers from unlabeled and unsegmented conversation, when the number of speakers is unknown. In this approach, Self-Organizing-Map (SOM) is used as likelihood estimators for speaker model. For estimation of the number of clusters the Bayesian Information Criterion (BIC) is applied. This approach was tested on the NIST 1996 HUB-4 evaluation test in terms of speaker and cluster purities. Results indicate that the combined SOM-BIC approach can lead to better clustering results than the baseline system.

1. Introduction

Most speaker recognition problems have been solved by using supervised methods. A less common problem is unsupervised speaker clustering, segmentation and indexing, where no labeled training data is available. The goal in this case is to assign the data to different clusters where each cluster represents a different speaker. Unlike most clustering approaches where each vector is associated with a specific cluster (static clustering), here a sequence of vectors has to be associated with the same cluster, which is known as a temporal data clustering.

Temporal data clustering has many applications including speaker recognition [1]-[7], machine monitoring [8], switching chaos [9], [10], prediction of systems output [9], clustering of EEG signals [10], music clustering [11], and protein modeling [12].

Temporal data clustering must be used when there is a successive dependence between data vectors in a group. An additional problem of temporal data clustering is to determine the change points (segmentation and change detection problems). Sometimes the transitions between the models are not sharp (e.g., one model appears before the end of the previous one) which is known as drifting dynamics [10]. In this case, it is necessary to find the transients and to give membership weights to each cluster at every time point.

Many approaches have been applied for temporal data clustering, e.g., the dendrogram [3]; the vector quantization (VQ) algorithms [4], the expectation maximization (EM) algorithm [5], [7]; hidden Markov model (HMM) [2], [6], [8], [12]; and neural networks (NN) [1], [9]-[11].

In the present study, a Code-Book (CB) is used to model each speaker. All the CBs are first trained such that each CB represents a different speaker. Then an iterative competitive algorithm is applied to all the CBs . The CBs are created applying a SOM algorithm [13]. In [1] each SOM was used to accumulate the Euclidian distance of a sequence. In this work each SOM was used as a likelihood estimator of a sequence. Input data was an unsegmented and unlabeled conversation, with unknown number of speakers, R . BIC was applied for the estimation of the number of speakers. This criterion has previously been applied to validate the speaker clustering for a Gaussian cluster model [7].

The next sections are organized as follows: Section 2 describes the proposed system. In section 3 we present the experiments and the results, and in section 4 the system and results are discussed.

2. Systems description

In general, given a conversation the goal is to estimate the number of clusters and to cluster the data into q clusters. The description of the VQ-based clustering system is summarized in subsection 2.1. Subsection 2.2 describes how the log-likelihood was estimated from the SOMs. In 2.3, the BIC validity criterion is presented. Sub-section 2.4 presents the SOM-BIC clustering system.

2.1. VQ-Based clustering system

Assuming that the number of speakers and the segment boundaries are known, and in each conversation the data includes, in addition to speech data, non-speech events, the goal of the algorithm is to cluster the input data into $R+1$ clusters. The initial conditions for the system were determined as follows: segments classified by the crude speech/non-speech classifier as non-speech were used to train the non-speech network. Segments classified as speech segments were randomly and equally divided and used to train the R speaker models.

For the following temporal-data clustering algorithm it is necessary to know the start and end points of each segment. In reality this information is not usually available. For this reason we cut the data into segments of fixed length (this length was set to one second, 100 frames). It was shown, [1] that 100 frames and a CB created using a SOM of 6×10 size are sufficient for speaker clustering.

The precise algorithm description and the proof of its convergence can be found in [1]. One iteration of the algorithm consists of the following three steps:

1. Retrain the models with the new partition achieved by the previous iteration.
2. Find a new attribution of each segment by finding the SOM with the minimal sum of the squares of the Euclidian distances.
3. Test for termination: if the termination criterion is met, exit; if not return to 1.

In the present work the following termination criterion was applied:

$$\frac{M_{change}}{M} \leq 0.01 \tag{1}$$

- M – Total number of segments.

- M_{change} – Number of the segments that change their attribution.

2.2. VQ as a log-likelihood estimator

As SOM-based CB is a Euclidian distance-based model and the log-likelihood must be calculated. The following approximation is applied: for input vector $v_n \in CB_r$ ($v_n \in \mathbb{R}^d$) we assume that each code-word, c_r^l, in the codebook is the mean of a Gaussian probability-density-function (*pdf*) with an identity covariance matrix. Then the estimated log-likelihood of one input vector is calculated as:

$$\begin{aligned} L\left(v_n \middle| c_r^{l^*}\right) &= -\frac{d}{2}\log(2\pi) - \frac{1}{2}\left(c_r^{l^*} - v_n\right)^T\left(c_r^{l^*} - v_n\right) \\ c_r^{l^*} &= \min_{l=1,\dots,L_r}\left\{\left(c_r^l - v_n\right)^T\left(c_r^l - v_n\right)\right\} \end{aligned} \quad (2)$$

Then the joint log-likelihood, for a model that consists of q codebooks, is estimated as:

$$\begin{aligned} L\left(\mathbf{V}\middle|\hat{\Theta}_q\right) &= -\frac{dN}{2}\log(2\pi) \\ &-\sum_{r=1}^{q}\sum_{v_n \in CB_r}\frac{1}{2}\left(c_r^{l^*} - v_n\right)^T\left(c_r^{l^*} - v_n\right) \end{aligned} \quad (3)$$

Where $\mathbf{V} = \{v_n\}_{n=1}^{N}$ and $\hat{\Theta}_q$ are the estimated parameters of all the codebooks, i.e., all the code-words $\left\{c_r^l\right\}_{r=1,\dots,q\,;\,l=1,\dots L_r}$.

2.3. The BIC validity criterion

The Bayesian Information Criterion for model selection was introduced by Schwarz in 1978 [14]. According to Schwarz, to select the best model for given data it is necessary to maximize the joint likelihood (log-likelihood $L(\mathbf{V},\hat{\Theta})$) of the data $\mathbf{V}$ and the estimated parameters $\hat{\Theta}$. According to the Bayes rule, $L(\mathbf{V},\hat{\Theta}) = L(\mathbf{V}|\hat{\Theta}) + L(\hat{\Theta})$. Schwarz showed that under the assumption of continuous parameters in some range, $L(\hat{\Theta})$ depends only on the number of estimated parameters $|\hat{\Theta}|$ and the number of data points that were used for parameter estimation, N. So the joint log-likelihood is:

$$L(\mathbf{V},\hat{\Theta}) = L(\mathbf{V}|\hat{\Theta}) - \frac{1}{2}|\hat{\Theta}|\log(N) \quad (4)$$

In practice the second term, often called a penalty term, is multiplied by a scaling factor λ to adjust the equation for a specific application. Then, a best system out of $R_{\max}$ estimated systems can be obtained by maximizing the joint log-likelihood using this scale factor.

$$R^* = \arg\max_{q=1,\dots,R_{\max}}\left\{L\left(\mathbf{V}\middle|\hat{\Theta}_q\right) - \frac{\lambda}{2}\left|\hat{\Theta}_q\right|\log(N)\right\} \quad (5)$$

The value of the scaling factor is task depended hyper-parameter. This shows that best results obtained by using $\lambda = 1.5$. However, there was no significant difference for $1.0 \le \lambda \le 2.5$.

2.4. SOM-BIC clustering system

By inserting (3) into (4) we get:

$$\begin{aligned} L\left(\mathbf{V},\hat{\Theta}_q\right) &= -\frac{dN}{2}\log(2\pi) \\ &-\sum_{r=1}^{q}\sum_{v_n \in CB_r}\frac{1}{2}\left(c_r^{l^*} - v_n\right)^T\left(c_r^{l^*} - v_n\right) - \frac{\lambda}{2}\left|\hat{\Theta}_q\right|\log(N) \end{aligned} \quad (6)$$

The feature vectors we used were LPCC of the dimension 12, and 60 code-words per one cluster model, i.e., each cluster has $|\Theta_{Cluster}| = |\Theta| = 60 \times 12 = 720$ parameters. The term $-\frac{dN}{2}\log(2\pi)$ is constant for all the systems for any value of q and can be discarded. The estimation of the number of clusters was therefore according to the minimization of the following expression:

$$\begin{aligned} R^* = \arg\min_{q=1,\dots,30}\Bigg\{&\sum_{r=1}^{q}\sum_{v_n \in CB_r}\left(c_r^{l^*} - v_n\right)^T\left(c_r^{l^*} - v_n\right) \\ &+|\Theta|q \cdot \lambda \cdot \log(N)\Bigg\} \end{aligned} \quad (7)$$

3. Systems evaluation

The system was tested on the NIST 1996 HUB-4 evaluation dataset. It is a broadcast news speech corpus, and the evaluation set consists of four datasets, each of approximately 30 minutes in duration. The four datasets are named File1, File2, File3, and File4 and the number of speakers is 7, 13, 15, and 20 respectively. The data was sampled at 16KHz, 16bits per sample, and was recorded from different channels, depends from the place that the journalist reported from.

3.1. Feature extraction

The features used were 12th order LPCC. The features calculated from 30ms frames with a 10ms frame rate. In addition, for system initialization, the mean absolute values for 50ms of accumulated frames were calculated for speech/non-speech evaluation. Preliminary segmentation of speech and non-speech data was performed by thresholding the absolute value feature. The threshold level was set at three percent of the maximum.

3.2. Evaluation criterion

Clustering evaluation was based on the purity concept explained in [5]. In [5], only Average Cluster Purity (acp) was calculated. This criterion encourages having over clustering, and in the extreme, one cluster per segment. The criterion used in this paper calculates the Average Speaker Purity (asp) as well, as was used in [2]. The reason for the second coefficient is to penalize the splitting of one speaker into several clusters. It is important to have a confidence measure taking both factors into account. In the case of speaker purity, non-speech data was ignored, because it is not relevant to relate it to a particulate cluster. The notation used is:

- R : Number of speakers

- q : Number of clusters
- n_{ij} :Total number of frames in cluster i spoken by speaker j
- $n_{.j}$:Total number of frames spoken by speaker j, $j=0$ means non-speech frames
- $n_{i.}$: Total number of frames in cluster i

The acp, based on cluster purity $\{p_{i.}\}_{i=0}^{q}$, can be defined as:

$$acp = \frac{1}{N}\sum_{i=0}^{q} p_{i.} \cdot n_{i.} \quad ; \quad p_{i.} = \sum_{j=0}^{R} \frac{n_{ij}^2}{n_{i.}^2} \tag{8}$$

Similarly, asp based on the speaker purity, $\{p_{j.}\}_{j=1}^{R}$, but without the non-speech data, is:

$$asp = \frac{1}{N - n_{.0}}\sum_{j=1}^{R} p_{j.} \cdot n_{.j} \quad ; \quad p_{.j} = \sum_{i=0}^{q} \frac{n_{ij}^2}{n_{.j}^2} \tag{9}$$

In order to compare between the systems, we calculate the evaluation criterion as the geometric mean of the acp and asp:

$$K = \sqrt{acp \cdot asp} \tag{10}$$

It is important to note that the values of acp, asp and K are always between zero and one independent of the number of speakers. Higher acp means that the cluster consists mostly of one speaker. Higher asp means that the speaker data does not split between many clusters. The optimal case is to maximize K, ideally with both acp and asp equal to one.

3.3. Experiment and results

As was shown in [1], a model with SOM size 6×10 (720 parameters per cluster) and 100 frames per segment are sufficient for speaker clustering. In all the experiments in this report we use these sizes.

The first row in Table 1 shows the results for the baseline system. In the baseline systems assumes that the number of speakers was known (R). Then the number of clusters (q) was set to $R+1$, where R is the number of speakers, plus an additional cluster for non-speech event, as in [1].

The second experiment includes clustering and estimation of the optimal number of clusters. The initial number was 30. At each stage the data was clustered and the validity was calculated, for $|\Theta| = 720$, according to (7). The scaling factor is a hyper-parameter that has to be found. Different values of scaling factor were applied: $\{\lambda_k\}_{k=0}^{6} = \{0.5k\}_{k=0}^{6}$. After validity calculation the cluster with the minimum amount of data was removed and the system was retrained with the reduced number of clusters. The process was continued until the number of clusters reduced to one. The penalty term influences the validation criterion, as bigger λ leads to a smaller number of clusters and vice versa. It was found that the best scaling factor was $\lambda = 1.5$. Table 1 shows the result of the clustering of the four files according to their scores:

- Second row: the score for the correct number of speakers, $R+1$ (one for non-speech events). It was done to compare the clustering performance of the baseline system with the full clustering system for the same number of clusters.
- Third row: the score for the best clustering result achieved according to the best K value as described in Section 3.2. This test helps to indicate how far is the system's clustering from the best result.
- Forth row: the score according to the estimated number of clusters, with penalty term $\lambda = 1.5$. This is the actual system performance row.

4. Conclusions

The temporal data clustering approach based on VQ, which was presented at [1], was applied for long conversations with different numbers of speakers. A maximum likelihood (ML) approach was used as a clustering criterion instead of a sum of Euclidian distances.

From the results analysis the following conclusions may be drawn:

1. The results for *a-priori* known number of speakers are the same as for clustering with the clustering reducing approach for $R+1$ clusters. This means that starting with a high number of clusters does not damage the clustering performance of the reduced number of clusters.
2. Results using the VQ-BIC approach are close to the best results, usually better than with $R+1$ clusters. As non-speech data can come from different sources, several clusters can be attached to these data. Speakers with close characteristics in the feature space can be attributed to the same model while speakers with high variability in their voice can be split into more than one cluster. For these reasons the optimal number of clusters may differ from $R+1$.
3. As the number of speakers increases, the performance of the system degrades. This is logical due to the fact that the number of estimated parameters that had to be estimated increases linearly with the number of speakers. Another reason is that as the number of speakers increases the overlapping between the clusters increases and the shapes that should be learned are more complex.

Estimation of the number of the participants (validity problem) is very important. In the presented validity criterion, the scaling factor for BIC is seems to be very important, in this application the results for $\lambda = 1.5$ and $\lambda = 2.0$ gave similar results.

The comparison of the results of this system with that described in [1] is as follows. The clustering results according to the ML approach and the minimum accumulated Euclidean sum are not statistically different. However, the ML approach is computationally less expensive. Comparing the validity criterion, the BIC criterion leads to better clustering performances. The drawback of the BIC criterion usually is in the hyper-parameter (λ) that is task dependent. However, it was shown that the clustering results in this study are robust to a various values of λ. Appling the criterion of [1] the systems always converge to a number of clusters not greater than five. This leads to a high asp and a low acp, i.e., several speakers in one cluster. Therefore, the overall clustering leads to a very low K.

Table 1: Clustering results for ($\lambda = 1.5$)

Model Type	File 1 – $R = 7$				File 2 – $R = 13$				File 3 – $R = 15$				File 4 – $R = 20$			
	N_c	acp	asp	K	N_c	acp	asp	K	N_c	acp	asp	K	N_c	acp	asp	K
Baseline	8	0.77	0.67	0.77	14	0.62	0.76	0.69	16	0.85	0.69	0.77	21	0.66	0.74	0.70
$R+1$ clusters	8	0.94	0.75	0.84	14	0.67	0.83	0.74	16	0.78	0.84	0.81	21	0.64	0.80	0.72
Best score	10	0.93	0.85	0.89	11	0.79	0.83	0.81	11	0.91	0.76	0.83	18	0.72	0.80	0.75
R^* clusters	10	0.93	0.85	0.89	10	0.79	0.79	0.79	11	0.91	0.76	0.83	11	0.80	0.71	0.74

5. Acknowledgment

The author wants to thank the Swiss Federal Office for Education and Science (OFES) in the framework of both EC/OFES "MultiModal Meeting Manager (M4) project" and the Swiss National Science Foundation, through the National Center of Competence in Research (NCCR) on "Interactive Multimodal Information Management (IM2)" for supporting this work.

6. References

[1] I. Lapidot (Voitovetsky), H. Guterman, and A. Cohen, "Unsupervised Speaker Recognition Based on Competition Between Self-Organizing-Maps," *IEEE Trans. on Neural Networks*, vol. 13, no.4, pp. 877-887, July 2002.

[2] J. Ajmera, H. Bourlard, and I. Lapidot, "Improved unknown-multiple speaker clustering using HMM," IDIAP, Martigny, Switzerland, Tech. Rep. IDIAP-RR02-23, August 2002.

[3] M. H. Kuhn, "Speaker recognition accounting for different voice conditions by unsupervised classification (cluster analysis)," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-10, no. 1, pp. 54-57, January 1980.

[4] A. Cohen and V. Lapidus, "Unsupervised text independent speaker classification," *Proc. of the Eighteenth Convention of Electrical and Electronics Engineers in Israel*, 1995, pp. 3.2.2 1-5.

[5] A. Solomonoff, A. Mielke, M. Schmidt, and H. Gish, "Clustering speakers by their voices," *ICASSP'98*, vol. 2, 1998, pp. 557-560.

[6] A. Cohen and V. Lapidus, "Unsupervised, text independent, speaker classification," *Proc. of the Int. Conf. on Signal Processing Application and Technology*, 1996, pp. 1745-1749.

[7] S. S. Chen and P. S. Gapalakrishnan, "Clustering via the Bayesian criterion with applications to speech recognition," *ICASSP'98*, vol. 2, 1998, pp. 645-648.

[8] L. M. D. Owsley, L. E. Atlas, and G. D. Bernard, "Self-organizing feature maps and hidden Markov models for machine-toll monitoring," *IEEE Trans. on Signal Processing*, vol. 45, no. 11, pp. 2787-2798, November 1997.

[9] K. Pawelzik, J. Kohlmorgen, and K.-R. Muller, "Annealed competition of expert for segmentation and classification of switching dynamics," *Neural Computation*, vol. 8, no. 2, pp. 340-356, February 1996.

[10] J. Kohlmorgen, K.-R. Muller, and K. Pawelzik, "Segmentation and identification of drifting dynamical systems," *Proc. Neural Networks for Signal Processing VII IEEE Workshop*, 1997, Amalia Island, USA, pp. 326-335.

[11] O .A. S. Carpinteiro, "A hierarchical self-organising map model for sequence recognition," *Pattern Analysis and Applications*, vol. 3, no. 3, pp. 289-287, 2000.

[12] A. Krogh, M. Brown, I. Saira Mian, K. Sjolander, and D. Haussler, "Hidden Markov models in computational biology applications to protein modeling," *J. Mol. Biol.*, vol. 235, no. 5, pp. 1501-1531, February 1994.

[13] T. Kohonen, "The self-organizing map," *Proc. IEEE*, vol. 78, no. 9, pp. 1464-1480, September 1990.

[14] G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461-464, 1978.