A NOVEL METHOD FOR SELECTING THE NUMBER OF CLUSTERS IN A SPEAKER DIARIZATION SYSTEM

Paula Lopez-Otero, Laura Docio-Fernandez and Carmen Garcia-Mateo

Multimedia Technologies Group (GTM), AtlantTIC Research Centre, Universidade de Vigo
EE de Telecomunicación, Campus Universitario de Vigo, 36310 Vigo

ABSTRACT

This paper introduces the cluster score (C-score) as a measure for determining a suitable number of clusters when performing speaker clustering in a speaker diarization system. C-score finds a trade-off between intra-cluster and extra-cluster similarities, selecting a number of clusters with cluster elements that are similar between them but different to the elements in other clusters. Speech utterances are represented by Gaussian mixture model mean supervectors, and also the projection of the supervectors into a low-dimensional discriminative subspace by linear discriminant analysis is assessed. This technique shows robustness to segmentation errors and, compared with the widely used Bayesian information criterion (BIC)-based stopping criterion, results in a lower speaker clustering error and dramatically reduces computation time. Experiments were run using the broadcast news database used for the Albayzin 2010 Speaker Diarization Evaluation.

Index Terms— Speaker Clustering, Cluster Similarity, Linear Discriminant Analysis

1. INTRODUCTION

Speaker clustering is a task consisting of grouping a set of speech segments in clusters. Each cluster must only include the speech segments of one speaker, and there must be only one cluster per speaker. Clustering is used in speaker diarization tasks, in which an audio stream is automatically segmented into speaker homogeneous segments. These segments are then clustered according to speaker identities [1]. Errors during the segmentation process influence the clustering task, due to the mis-classification of speech and non-speech and the removal of speaker change-points.

In speaker clustering, it is common to represent the speech segments as supervectors obtained by concatenating the means of an adapted universal background model (UBM) [2]. Subspace projection techniques are widely used in speaker identification, in order to improve the separability of the different classes and to reduce the dimensionality of the data. The different subspace projection techniques can be divided into supervised and unsupervised techniques. Among the supervised techniques, a classic approach is linear discriminant analysis (LDA) [3] and variants, such as kernel LDA [4] and probabilistic LDA [5]. The unsupervised techniques include principal component analysis [6] and the factor analysis-based iVector approach [7]. The use of supervised techniques for speaker clustering is not straightforward due to the fact that the number of speakers (clusters) and their identities are unknown. Thus, only unsupervised or partially supervised projection techniques can be used.

The literature refers to several approaches to the speaker clustering task: the classic Bayesian information criterion (BIC) approach [8], the information change rate (ICR) [9] and the generalized likelihood ratio (GLR) [10], among others. Nevertheless, how to decide the number of clusters has not been satisfactorily addressed.

In this work, a method for selecting the number of clusters called cluster score (C-score) is presented, which is competitive both in terms of computational efficiency and performance. It consists of a measure that finds a trade-off between intra-cluster and extra-cluster similarities [11]. The C-score is combined with a partially supervised projection technique based on LDA.

The rest of the paper is organized in sections that describe the following: Section 2, feature representation and the LDA projection technique; Section 3, the clustering strategy and the C-score technique for selecting the number of clusters; Section 4, the database and the metrics used for assessing the diarization system; Section 5, our experimental results; Section 6, conclusions and future research.

2. FEATURE REPRESENTATION

Given an audio stream which was segmented into a set of n_s speech segments $S = (S_1, \ldots, S_{n_s})$ the following steps are applied:

- First, acoustic features are extracted from the waveform. Specifically, 12 mel-frequency cepstral coefficients (MFCCs) and normalized log-energy are extracted every

This work has been supported by the European Regional Development Fund, the Galician Regional Government (CN2011/019, 'Consolidation of Research Units: AtlantTIC Project' CN2012/160) and the Spanish Government (FPI grant BES-2010-033358 and 'SpeechTech4All Project' TEC2012-38939-C03-01).

10 ms using a 25 ms Hamming window, and augmented with first and second order dynamic coefficients resulting in a feature vector of dimension N. Cepstral mean and variance normalization are also applied.

- For each segment S_i, a UBM of R mixtures is adapted to its corresponding acoustic features using the maximum a posteriori (MAP) algorithm. As a result, a set of R adapted mean vectors of N features is obtained, and these means are concatenated forming a supervector $\mathbf{v_i}$ of dimension $D = RN$.
- The set of segments is now represented by means of matrix $\mathbf{V} = (\mathbf{v_1 v_2} \cdots \mathbf{v_{n_s}})$, where the i^{th} column of $\mathbf{V}$ is the supervector that represents segment S_i.

The feature representation described above represents a set of speech segments by means of a matrix $\mathbf{V}$, where each column is the supervector corresponding to one speech segment or, equivalently, each column of $\mathbf{V}$ is a point in a reference space defined by the UBM. The aim of the clustering task is to group these points into homogeneous classes, and to do so, it is important that the points belonging to the same class are close to each other and, at the same time, far from the points belonging to the other classes. Thus, in this work, LDA is applied to the supervectors in order to reduce their dimensionality while increasing the separability of the different classes [3]. By training a transformation matrix $\mathbf{X}$, the original data can be projected into a more discriminative subspace as follows:

$$\mathbf{V_{LDA}} = \mathbf{V^T} \cdot \mathbf{X} \tag{1}$$

where $\mathbf{X} \in \Re^{D \times D_{LDA}}$ and $\mathbf{V_{LDA}}$ is a matrix whose i^{th} column represents the speech utterance S_i in a discriminative space where the original supervector $\mathbf{v_i}$ of dimension D is now a supervector $\mathbf{v_{LDA_i}}$ of dimension D_{LDA}, with $D_{LDA} < D$.

A procedure to train $\mathbf{X}$ must be defined. Supervised training is ruled out for this application since the number of speakers is not known a priori. Thus, the partially supervised training proposed in [12] is performed: a speaker discriminative transformation matrix $\mathbf{X}$ is trained in a training dataset, where the number of speakers is known, and this transformation is then applied to the test dataset.

3. SPEAKER CLUSTERING STRATEGY

Once the speech segments are represented by a matrix $\mathbf{V}$ ($\mathbf{V_{LDA}}$) as described in Section 2, speaker clustering is performed. Agglomerative hierarchical clustering (AHC) is used in this work for this task, due to its simplicity and acceptable results [9]. In this algorithm, each speech segment initially constitutes a cluster in itself. The distance between clusters is computed and the most similar pairs are merged. This process is repeated until a stopping criterion is met [13].

As described in Section 2, each column of $\mathbf{V}$ ($\mathbf{V_{LDA}}$) is a vector in a reference space defined by the UBM. Thus, the similarity between pairs of speech segments can be computed straightforwardly using the cosine similarity between the two corresponding vectors. As stated in [12], the representation of the information by means of supervectors shows very strong directional scattering patterns, which makes the direction of the points more informative than their magnitude. AHC decides whether to merge two clusters by means of a merge criterion. Experiments with the database used in this work were run to decide which merge criterion to use. After assessing the single-link, complete-link, average-link and Ward's methods [14], results showed that the merge criterion that performed best was the average-link approach, in which the distance between two clusters is the average of all pairwise distances between the elements in the two clusters.

The clustering toolkit CLUTO [15] was used for this stage. The segments were clustered until a sole cluster was formed, resulting in a dendrogram that was cut at different nodes. Thus, a set of clustering solutions $C = (C_1, \ldots, C_{n_s})$ was obtained.

3.1. Selecting the number of clusters

We present a method to decide the number of clusters at the clustering stage based on intra-cluster and extra-cluster similarities. These concepts have to be introduced before defining the strategy to select the number of clusters. Hence, given a clustering solution $C_n = (c_1, \ldots, c_n)$ with n clusters, where cluster c_i has $\#c_i$ elements and where these elements are supervectors as described in Section 2, I_n represents similarity between elements in the same cluster and E_n represents average similarity of the elements in a cluster and the rest of the elements in other clusters:

$$I_n = \frac{1}{n_s} \sum_{k=1}^{n} \#c_k \left(\frac{1}{\#c_k^2} \sum_{\mathbf{v_i}, \mathbf{v_j} \in c_k} cos(\mathbf{v_i}, \mathbf{v_j}) \right) \tag{2}$$

$$E_n = \frac{1}{n_s^2 - \sum_{k=1}^{n} \#c_k^2} \sum_{k=1}^{n} \left(\sum_{\substack{\mathbf{v_i} \in c_k \\ \mathbf{v_j} \notin c_k}} cos(\mathbf{v_i}, \mathbf{v_j}) \right) \tag{3}$$

where $\mathbf{v_i}$ and $\mathbf{v_j}$ are the supervectors of segments i and j, respectively. I_n is the mean of the average of all the cosine similarities between the elements assigned to the same cluster, and E_n is the sum of all the cosine similarities between each element in a cluster and the elements of the other clusters, weighted according to the distribution of the cluster sizes. It should be noted that $n_s = \sum_{k=1}^{n} \#c_k$.

I_n and E_n are computed by the CLUTO toolkit [11], and their values range between -1 and 1.

We propose an approach for finding a clustering solution C_n which maximizes I_n and minimizes E_n (or equivalently, maximizes $1 - E_n$). Since in real-world scenarios, as I_n increases $1 - E_n$ decreases and vice versa, a trade-off between I_n and $1 - E_n$ must be achieved. This can be accomplished

by using the harmonic mean of $\hat{I}_n$ and $1-\hat{E}_n$, which we have named $C\text{–}score_n$:

$$C\text{–}score_n = \frac{2\hat{I}_n(1-\hat{E}_n)}{\hat{I}_n + (1-\hat{E}_n)} \quad (4)$$

where $\hat{E}_n$ and $\hat{I}_n$ are mapped versions of E_n and I_n to the interval $[0, 1]$.

Thus, our method selects a clustering solution C_{n^*} with n^* clusters, where n^* is chosen as follows:

$$n^* = \underset{i=n_{min},\ldots,n_{max}}{\arg\max} \; C\text{-}score_i \quad (5)$$

The clustering procedure is summarized in Algorithm 1.

As can be seen in Eq. 5, the possible values of n^* do not range from 1 to n_s but from n_{min} to n_{max}, with $n_{min} > 1$ and $n_{max} < n_s$. This constraint is applied in order to avoid over-clustering (almost all the segments would be in the same cluster) or under-clustering (almost all the segments would form a cluster on their own).

n_s greatly varies depending on the data to be clustered; thus, selecting a fixed value for n_{min} and n_{max} would result in a satisfactory performance in some concrete situations but a poor performance in others. Thus, it is proposed to select n_{min} and n_{max} in function of n_s as follows: $n_{min} = \frac{n_s}{k_c}$, $n_{max} = \frac{2n_s}{k_c}$. where k_c is a constant. In this way, n_{min} and n_{max} are in function of n_s and there is only one control parameter k_c, which has to be tuned using development data.

Algorithm 1 AHC clustering with C-score

Require: Speech segments $S = (S_1, \ldots, S_{n_s})$
1: MAP adaptation of UBM and concatenation of means $\rightarrow (\mathbf{v_1}, \ldots, \mathbf{v_{n_s}})$
2: Matrix $\mathbf{V}$ | column $v_{*,i} = \mathbf{v_i}$
3: Transformation of $\mathbf{V}$: $\mathbf{V_{LDA}} = \mathbf{V} \cdot \mathbf{X}$
4: AHC of matrix $\mathbf{V}$ ($\mathbf{V_{LDA}}$) $\rightarrow$ clustering solutions $C = (C_1, \ldots, C_n)$
5: **for** $i = n_{min} \rightarrow n_{max}$ **do**
6: Compute C-score of $C_i \rightarrow C\text{-}score_i$
7: **end for**
8: **return** $C_{n^*} \mid n^* = \underset{i=n_{min},\ldots,n_{max}}{\arg\max} \; C\text{-}score_i$

4. EXPERIMENTAL FRAMEWORK

The broadcast news database employed in Albayzin 2010 SDE [16] was used to assess the performance of the proposed clustering approach. This database consists of broadcast news programmes recorded from the Catalan 3/24 TV channel. It is split into 24 sessions, 16 for development (57.5 hours) and 8 for testing (30 hours). The number of speakers per session ranges from 30 to 250 (mean, 85; standard deviation, 24) per session. The development dataset was used to adjust the free parameters and to train the UBM and the LDA transformation matrix $\mathbf{X}$. The test dataset was used to assess the performance of the different approaches.

Two scenarios were assessed in this work:

- Scenario with no segmentation errors (namely manual segmentation): the set of manually segmented speaker turns included in Albayzin 2010 SDE database was used in this scenario. n_s ranged from 150 to 1100 segments per session.
- Scenario with segmentation errors (namely automatic segmentation): a set of automatically segmented speaker turns was obtained by a BIC-based automatic segmentation system that models the occurrences of change-points by means of a Poisson process [17]. n_s ranged between 100 and 630 segments per session, the false alarm (FAS, non-speech labelled as speech) rate was 2.2% and the missed speech (MS, speech labelled as non-speech) rate was 7.3%, as shown in Table 1.

The metric used to assess the system was the time-based speaker diarization error score (SPKE) [18], which is the percentage of speech incorrectly assigned to a speaker. This metric reflects the amount of speech that was assigned to a wrong speaker after optimal mapping of the automatically assigned speakers and the reference speakers.

5. EXPERIMENTAL RESULTS

In order to assess the results obtained in selecting the number of clusters with the proposed C-score technique, the same experiments were run following the classical BIC-based approach described in Section 5.1 below. Also, the performance of the C-score and BIC approaches was compared with the performance ceiling, which is the lowest possible SPKE. This performance ceiling would be obtained if the number of clusters that achieved the lowest SPKE were chosen in every case, and represents a scenario where there are no errors when selecting the number of clusters.

5.1. Reference system

In the BIC-based stopping criterion strategy [9], a value ΔBIC was computed every time two clusters i and j were about to be merged:

$$\Delta BIC(i,j) = L(i,j) - \lambda P \quad (6)$$

where $L(i,j)$ represents the likelihood of merging clusters i and j minus the likelihood of not merging them, P is a penalty corresponding to the number of free parameters in the model, and λ is a free parameter. When $\Delta BIC > 0$ the clusters were similar, but when $\Delta BIC < 0$, the clusters were not alike enough to be merged, so clustering stopped at that point. It must be noted that λ has to be tuned in order to adjust the threshold that will cause the algorithm to stop clustering [19], as low values of λ lead to a premature stopping of the clustering procedure (resulting in too many clusters), and high values of λ cause the algorithm to cluster data until too few clusters are formed.

5.2. Parameter tuning

The feature vectors used in these experiments were of dimension $N = 39$ and the UBM used to obtain the supervectors was a Gaussian mixture model with $R = 64$ mixture components, leading to $D = RN = 2496$-dimensional mean supervectors. The LDA transformation described in Section 1 projected the mean supervectors into vectors of dimension $D_{LDA} = 200$, which was the subspace dimensionality that achieved the lowest SPKE in development.

Several parameters were tuned using the development data: those of the speaker segmentation system and also the free parameter of the two strategies for selecting the number of clusters. The values of these free parameters that achieved the lowest SPKE were selected. In the case of the C-score strategy, the value selected for k_c was 12, and in the case of the BIC strategy, a different value of λ was selected for each set of speaker turns and for each utterance representation: $\lambda = 48, 50$ for manually segmented speaker turns and $\lambda = 19, 35$ for automatically segmented speaker turns with the supervector and LDA representations, respectively.

5.3. Discussion

Table 1 shows the results obtained when clustering the manually and automatically segmented speaker turns using the C-score and the BIC. The C-score technique obtained a lower SPKE than the BIC when representing the data with the mean supervector approach (SV) and with the LDA projection. Moreover, these SPKE were close to the performance ceiling. The LDA representation achieved the best diarization results, and it also had the lowest performance ceiling. It must be noted that, despite the segmentation errors, SPKE was very similar for the manually and automatically segmented speaker turns. This means that the clustering technique is robust to speaker segmentation errors.

Table 1. Diarization results and 95% confidence interval for manually and automatically segmented speaker turns.

Segmentation	Method	FAS	MS	SPKE (%)	
				SV	LDA
Manual	C-score	0%	0%	**22.1± 1.0**	**16.1 ± 0.9**
	BIC			27.4 ± 1.1	29.0 ± 1.1
	Ceiling			**20.7 ± 1.0**	**13.6 ± 0.8**
Automatic	C-score	2.2%	7.3%	**19.7 ± 0.8**	**15.0 ± 0.7**
	BIC			21.6 ± 0.8	19.4 ± 0.8
	Ceiling			**15.8 ± 0.7**	**12.7 ± 0.7**

The C-score method combined with the subspace projection technique outperformed other diarization systems that used the same database, as can be confirmed in [16] and [20].

Measurements of the computation time of the techniques employed in this work were taken, finding that the real-time factor for the C-score was around 10^{-6}, but was around 10^{-3} in the case of BIC. These measurements also revealed that the computation time of the BIC approach was dependent on the value of λ: the lower λ, the faster the algorithm.

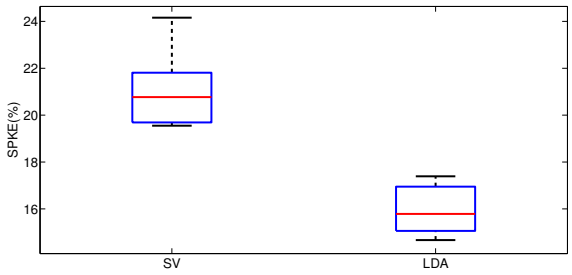

Fig. 1. Boxplots of the SPKE obtained with $k_c = 9, \ldots, 14$. The red mark is the median, the edges of the box are the 25^{th} and 75^{th} percentiles and the whiskers extend to the most extreme datapoints.

As mentioned in Section 3.1, the C-score technique has a tuning parameter k_c that decides the range of possible values for the number of clusters (n_{min} and n_{max}). This parameter can be tuned using a development dataset, but it is important to study its sensitivity. Experiments with values of $k_c = 9, \ldots, 14$ were run, and results showed that the variation of SPKE in the SV representation ranged from 19% to 24%, but was much slighter in the case of the LDA representation, as it ranged from 15% to 17%. Thus, this subspace projection technique gives robustness to the C-score algorithm, as the sensitivity to its free parameter is reduced. Figure 1 represents boxplots showing the SPKE of the test dataset obtained with different values of k_c for the different data representations. These boxplots confirm that the subspace projection technique improve the performance of the clustering stage with respect to the SV representation. This LDA projection also show less sensitivity to the selection of k_c than in the case of the SV representation. Thus, the LDA projection technique give robustness to the C-score algorithm, as the sensitivity to its free parameter is reduced.

6. CONCLUSIONS AND FUTURE WORK

This paper introduces the C-score measure for selecting the number of clusters in a speaker diarization system that achieves a trade-off between maximizing intra-cluster similarity and minimizing extra-cluster similarity. This measure is assessed when speech utterances are represented by a GMM mean supervector and when this supervector is projected into a discriminative subspace by applying LDA. Compared with the well-known BIC-based approach, experimental results showed that the C-score combined with the LDA projection technique obtained an improvement in performance and a reduction in computation time, proving as well to be robust

to segmentation errors, as results were not degraded when clustering automatically segmented data. Also, this approach proved to have little sensitivity to its free parameter k_c, which gives this algorithm a value added with respect to the BIC algorithm, sensitive to its penalty weight λ. Furthermore, the tuning for λ takes longer than for k_c.

The performance shown by this clustering approach is promising, but there is still room for improvement. The performance ceiling, which is dependent on the clustering strategy, has not been reached. The C-score strategy will be assessed with different clustering approaches in order to improve the results presented here. We also plan to study techniques for fusing different clustering solutions, in order to assess whether combining the approaches presented in this work with other strategies leads to a better clustering solution.

REFERENCES

[1] S. E. Tranter and D. A. Reynolds, "An overview of automatic speaker diarization systems," *Trans. Audio, Speech and Lang. Proc.*, vol. 14, no. 5, pp. 1557–1565, Sept. 2006.

[2] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *IEEE Signal Processing Letters*, vol. 13, pp. 308–311, 2006.

[3] Qin Jin and Alex Waibel, "Application of LDA to speaker recognition," in *Proceedings of the ICSLP*, 2000, pp. 250–253.

[4] G. Baudat and F. Anouar, "Generalized discriminant analysis using a kernel approach," *Neural Computation*, vol. 12, no. 10, pp. 2385–2404, Oct. 2000.

[5] S.J.D. Prince and J.H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *IEEE 11th International Conference on Computer Vision, ICCV 2007*, 2007, pp. 1–8.

[6] O. Thyes, R. Kuhn, P. Nguyen, and J.-C. Junqua, "Speaker identification and verification using eigenvoices," in *Proceedings of ICSLP*, 2000, pp. 242–245.

[7] N. Dehak, P.J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech and Language Processing*, 2010.

[8] S. S. Chen and P. S. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the Bayesian information criterion," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, 1998, pp. 127–132.

[9] K. J. Han and S. Narayan, "A robust stopping criterion for agglomerative hierarchical clustering in a speaker diarization system," in *Proceedings of Interspeech*, 2007, pp. 1853–1856.

[10] K. J. Han, S. Kim, and S. S. Narayanan, "Robust speaker clustering strategies to data source variation for improved speaker diarization," in *Proceedings of ASRU*, 2007, pp. 262–267.

[11] Y. Zhao and G. Karypis, "Evaluation of hierarchical clustering algorithms for document datasets," in *Proceedings of 11th International Conference of Information and Knowledge Management (CIKM)*, 2002, pp. 515–524.

[12] H. Tang, S.M. Chu, M. Hasegawa-Johnson, and T.S. Huang, "Partially supervised speaker clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 5, pp. 959–971, May 2012.

[13] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, 1999.

[14] William B. Frakes and Ricardo Baeza-Yates, Eds., *Information retrieval: data structures and algorithms*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1992.

[15] George Karypis, "CLUTO - a clustering toolkit," Tech. Rep. #02-017, Nov. 2003.

[16] M. Zelenák, H. Schulz, and J. Hernando, "Albayzin 2010 evaluation campaign: Speaker diarization," in *Proceedings of FALA 2010 - VI Jornadas en Tecnología del Habla and II Iberian SLTech Workshop*, 2010, pp. 301–304.

[17] P. Lopez-Otero, L. Docio-Fernandez, and C. Garcia-Mateo, "Novel strategies for reducing the false alarm rate in a speaker segmentation system," in *Proceedings of ICASSP*, 2010, pp. 4970–4973.

[18] "The NIST rich transcription evaluation project website," `http://www.itl.nist.gov/iad/mig/tests/rt/`.

[19] J. Žibert and F. Mihelič, "Novel approaches to speaker clustering for speaker diarization in audio broadcast news data," *Speech recognition: technologies and applications. Artificial intelligence series*, pp. 341–362, 2008.

[20] M. Diez, M. Penagarikano, A. Varona, L.J. Rodriguez-Fuentes, and G. Bordel, "On the use of dot scoring for speaker diarization," in *Proceedings of the 5th Iberian conference on Pattern recognition and image analysis*, 2011, IbPRIA'11, pp. 612–619.