ESANN 2010 proceedings, European Symposium on Artificial Neural Networks - Computational Intelligence and Machine Learning. Bruges (Belgium), 28-30 April 2010, d-side publi., ISBN 2-930307-10-2.

Online Speaker Diarization with a Size-Monitored Growing Neural Gas Algorithm

Jean-Louis Gutzwiller, Hervé Frezza-Buet and Olivier Pietquin *

SUPELEC - Metz Campus - IMS Research group
2 rue Edouard Belin - 57070 Metz, FRANCE
firstname.lastname@supelec.fr

Abstract. This paper proposes a method for segmenting and clustering an audio flow on the basis of speaker turns. This process, also known as speaker diarization, is of major importance in multimedia indexation. Here, we propose to realize this process online and without any prior knowledge on the number of speakers. This is done thanks to a statistical modelling of speakers based on a size-monitored growing neural gas algorithm.

1 Introduction

Speaker diarization is a process by which a machine automatically splits and labels an audio stream into homogeneous regions according to speakers. It has very important applications and is often a prerequisite to multimedia material indexation, automatic subtitling or active help for ear-impaired people. In the most challenging cases, nothing is known about the speakers (no model), their number nor the number of appearance of each particular speaker. Also, no prior knowledge is available about the quality of the audio material. Usually, the assumption that speakers never talk simultaneously is made. The diarization process can therefore be split into two distinct phases. First, speaker turns have to be detected, that is the boundaries of homogeneous speech segments (ideally as long as possible) are identified. Second, a clustering phase arises to assign each segment to a speaker label in an unsupervised manner.

Most often, even if the first phase is usually done online, the second occurs off-line and the full audio material has to be recorded before clustering can be applied [1, 2]. Very few online methods can be found in the literature [3]. This limitation prevents *anytime* use of indexation engines involving speaker diarization. Anytime applications are nevertheless more and more common and especially in the domain of audio-visual surveillance systems or automatic generation of meeting minutes [2] for instance. In this paper, we propose an algorithm based on statistical modelling and topological neural networks to perform online speaker diarization.

The rest of this paper is organized as follows. Section 2 explains how speech is pre-processed so as to extract meaningful features. Section 3 describes the splitting procedure. Section 4 describes the topological-neural-network-based method. Section 5 provides experimental results. Finally, Section 6 concludes.

*This work has been supported by the EUREKA project LINDO (ITEA2 (06011)).

ESANN 2010 proceedings, European Symposium on Artificial Neural Networks - Computational Intelligence and Machine Learning. Bruges (Belgium), 28-30 April 2010, d-side publi., ISBN 2-930307-10-2.

2 Feature Extraction

So as to reduce the amount of data to be processed and take advantage of natural redundancy, features are extracted from the speech signal. A lot of features are known in the field of speech processing but among them, *Mel Frequency Cepstrum Coefficients* (MFCC) [4] are know to perform well. The MEL[1] analysis technique provides weakly correlated coefficients which are meaningful parameters for speech recognition but also speaker identification [5]. We therefore chose to extract 24 MFCCs from 32ms sliding analysis windows overlapping by 22ms. They constitute the 16 components of a so-called acoustic vector or feature vector. An acoustic vector is thus extracted every 10ms.

$$s[n]_{n\in[0,255]} \xrightarrow{FFT} S[j]_{j\in[0,255]} \xrightarrow{MEL} S[k]_{k\in[0,15]} \xrightarrow{\log||} \log|S[k]|_{k\in[0,15]} \xrightarrow{IDTC} c[i]_{i\in[0,15]}$$

Fig. 1: MFCC computation

MFCC coefficients are obtainned as described in Fig.1. First, a Fast Fourier Transform (FFT) is applied to N acoustic samples ($s[n]_{n\in[0,N-1]}$) so as to obtain the N coefficients of the speech spectrum ($S[j]_{j\in[0,N-1]}$) (zero-padding could be used to increase the number of spectrum coefficients). Typically, speech is sampled at 8 kHz and 32ms represent approximatively 256 samples ($N = 256$). Secondly, the MEL-spectrum vector ($S[k]_{k\in[0,15]}$) is obtainned by feeding a MEL filter bank with the magnitude of the spectrum. The MEL filter bank actually approximates the non-linear human hearing process by transforming the frequency scale into a non-linear scale [6]. The reduction in coefficient number (from N to 16) arises from the filtering of this transformed spectrum. A *logarithmic* transform followed by an *Inverse Discrete Cosine Transform* (IDCT) finally provides the *MEL-cepstrum* (or MFCC) coefficient vector ($c[i]_{i\in[0,15]}$) for the current window.

3 Speaker Turns Detection

Although several methods can be cited to realize this splitting procedure (see [2], Section 2.2), a distance-based method is chosen here. Speaker changes will be detected as maxima of a distance computed between contiguous sets of acoustic vectors similarly to [7]. To compute this distance, two sliding windows are defined over acoustic vectors. Each window contains 300 acoustic vectors representing 3 seconds of speech. If we suppose that the acoustic vectors are drawn from a 16-dimensional Gaussian distribution, determining a distance between two sets of acoustic vectors reduces to the computation of a statistical distance between two Gaussian distributions whose parameters have to be estimated for

[1] The name comes from the word *melody*

each window. Notice that this implicitly means that a speaker has to speak during at least 3 seconds at each turn. Many statistical distances may be proposed [8] but we only considered the Mahalanobis distance (d^{12}_{MAH}) and a normalized version of the Euclidian distance (d^{12}_{NOR}) defined by (the latest being equivalent to the Mahalanobis distance if the variance-covariance matrix is diagonal):

$$d^{12}_{MAH} = \frac{1}{n}(\bar{\mu}_2 - \bar{\mu}_1)^T(\Sigma_1\Sigma_2)^{-1}(\bar{\mu}_2 - \bar{\mu}_1)$$

$$d^{12}_{NOR} = \frac{1}{n}\sum_{i=0}^{15}\frac{(\mu_2 - \mu_1)^2}{\sigma_1\sigma_2}$$

where $\bar{\mu}_1$ and Σ_1 (resp. $\bar{\mu}_2$ and Σ_2) are the mean vector and the covariance matrix of acoustic vectors in the first window (resp. second window).

The distance is computed every 500ms meaning that the two analysis windows are slided of 500ms and that statistics are recomputed at each step. This provides a discrete distance signal from which maxima have to be detected online. From experiments, this distance signal being actually quite noisy a low-pass filter is first applied to avoid extra detections. The maxima detection is threshold-based. The first assumption is that two consecutive maxima cannot be found within a temporal window of width t (this means that one person speaks at least t seconds, typically 20s). Second, the absolute maximum in a t-width window should be above a given absolute threshold α (automatically computed as a multiple of the standard deviation σ of the distance computed so far). Finally, there should be a number p of distances within the temporal window that are below an other threshold β automatically computed as a proportion of the detected maxima magnitude. In other words, in a window of width t, we select the absolute maxima if it is above α and if it at least $p\%$ of the other distances are below β (α and β being computed automatically). We kept this speaker turn detection computationally simple because the novelty of this paper resides in the clustering part which is explained hereafter. Moreover, false detections can be recovered to a certain extent by an appropriate clustering method able to merge consecutive segments if they where uttered by a same speaker.

4 Speaker Clustering

The originality of this paper resides in the speaker clustering method and especially because on-line computation is strongly highlighted. This implies managing signals as streams. The current speaker is characterized by the current distribution of the acoustic vectors he/she produces. This distribution may slightly change as one person speaks, but is supposed to change more significantly when the speaker changes. Tracking such non-stationary distribution, dealing with smooth changes as well as dramatic ones, has been tackled in the framework of video processing by a suitable vector quantization technique [9]. This technique, named GNG-T (for Growing Neural Gas with Target), is used here.

ESANN 2010 proceedings, European Symposium on Artificial Neural Networks - Computational Intelligence and Machine Learning. Bruges (Belgium), 28-30 April 2010, d-side publi., ISBN 2-930307-10-2.

GNG-T consists in quantifying a distribution with a finite set of prototypes, as with the k-means algorithm [10]. The main differences are twofold. First, GNG-T prevents from setting the number of prototypes in advance, but rather adjusts this number so that the quantification error is kept constant to a predefined target. During distribution changes, the number of prototypes may vary dynamically, thus avoiding both oversampling and undersampling of the distribution. The second difference with k-means is that prototypes are structured as graph, that reflects the "shape" of the distribution, see [11] for details. The setting up of this graph has been adapted from GNG algorithm [12]. This graph is not explicitly used here, but it gives GNG-T the ability to adapt quickly to smooth as well as abrupt distribution evolutions, the latter occurring when the current speaker changes.

Clustering is thus done as follows. A GNG-T algorithm is fed with acoustic vectors (MFCC) of the last time segment which boundaries were found during the speaker turn detection phase. The use of GNG-T provides a graph that represents the MFCC distribution of the current speaker. Once computed, each graph, corresponding to one segment, is stored. After each graph computation, the current graph is compared to the others.

To do so, we need a comparison method between graphs. Here again, we will use a distance-based method. A distance between graphs has therefore to be defined. There exist several metrics in the literature [13] and several were tried. We only report here the choice that resulted in better performances. The chosen method consists in associating each node of one graph the closest node in the other graph and to sum the distances of the formed pairs. This distance is commutative.

The clustering procedure is then as follows. The current graph is compared in terms of distance with the graphs extracted from previous segments. If the distance is above a given threshold δ (typically quite low), the segment is not eligible for being associated to the same speaker. The mean distance between all the non-eligible segments is then computed and provides an information about the mean distance between different speakers. A second threshold γ is computed as a multiple of this mean distance and serves to associate the current speaker to a previous segment. If there is no segment for which the distance is below γ, than a new speaker label is created. Since each graph is computed once for each segment and possesses a few nodes compared to the number of initial acoustic vectors, the distance computation can be done online in real time even for a large number of segments.

5 Experiments

Experiments have been conducted on the BREF80 database [14]. This database contains several hours of French read-speech spoken by 90 different native speakers (50 female and 40 male speakers). Eleven speakers were randomly selected in this database. A 1-hour and a 10-hour audio files were then created by concatenating segments randomly selected in the audio material corresponding to

these 12 speakers. The segment duration ranges from 10 to 40 seconds and the test databases contains respectively approximately 150 and 1500 segments. Several measures were computed. First, after the speaker-turn detection stage, the sensibility and the purity were computed as follow :

$$Sensibility(\%) = \frac{TP}{TP + FN} \times 100$$

$$Purity(\%) = \frac{\text{time of main speaker in the segment}}{\text{total time of the segment}} \times 100$$

where TP stands for *true positives* (the correctly detected transitions) and FN stands for *false negatives* (detections where there is no speaker-turn change). Thus, we try to reach high values of both the sensibility and the purity measures which will be obtainned for correct detections and longer homogeneous segments. Results are displayed in Table 1.

	Sensibility	Purity
DB 1h	96.0 %	97.3 %
DB 10h	96.7 %	96.6 %

Table 1: Results of speaker-turn detection

	Match. (N)	Match. (T)	Purity	Comp. Purity	n Speakers
DB 1h	97.0 %	98.0 %	1.03	1.03	12
DB 10h	95.1 %	96.8 %	1.03	1.02	11

Table 2: Results of speaker clustering

Then, the clustering method is evaluated. To do so, several metrics are used. First, the *matching* measures compute the percentage (in terms of global number (N) or in terms of time duration (T)) of correctly matched segments (correctly labelled segments after clustering according to the ground truth). The *Purity* is computed as before but replacing the main speaker in the segment by the main speaker in the corresponding segment of the ground truth. The *Comp. Purity* measure is complementary to the *Purity* measure and is computed as the ratio of the time used by the main speaker of the cluster (given by the automatic method) in the corresponding segment in the ground truth. This measure is mandatory to assess the quality of the clustering because a good *purity* value can be reached by over segmentation but will produce a bad *comp. purity* value. A good labelling will therefore lead to high *matching* values as well as *purity* and *comp. purity* values close to 1.

Results are displayed in Table 2 together with the number of detected speakers (*n Speakers*) (remembering that there are actually 11 speakers). It can be seen that the results are very good (12 or 11 speakers are found, the segments are almost pure and the matching is accurate) and similar to the state of the art [1]. Remember than unlike most of the state-of-the-art methods, our algorithm is online. Notice that results of Table 1 can be improved by merging consecutive clusters sharing the same label.

6 Conclusion

In this paper we proposed a new online speaker diarization algorithm. This algorithm is split into two phases : a splitting procedure and a clustering procedure. The novelty of this paper resides essentially in the clustering procedure. This is based on the modeling of the distribution of acoustic features for a given speaker as a graph with a reduced number of nodes compared to the initial distribution. This graph is obtainned thanks to the GNG-T algorithm and the speakers are compared according to an inter-graph distance measure. This distance-based algorithm can be performed online since it only works on small graphs. Moreover, it also doesn't require any model of the user or any prior information about the number of speakers unlike most of the state-of-the-art algorithms. Performance of the algorithm has been assessed on a standard speech database. Experimental results show that the proposed method reaches fairly good results comparable to the state-of-the-art although being online and totally unsupervised.

References

[1] P. Delacourt and C. Wellekens. Distbic: A speaker-based segmentation for audio data indexing. *Speech Communication*, 32(1-2):111–126, September 2000.

[2] X. Anguera. *Robust Speaker Diarization for Meetings*. PhD thesis, Universitat Politècnica de Catalunya, Barcelona, October 2006.

[3] J. Rougui, M. Rziza, D. Aboutajdine, M. Gelgon, and J. Martinez. Fast incremental clustering of gaussian mixture speaker models for scaling up retrieval in on-line broadcast. In *Proc. of ICASSP 2006*, Toulouse, 2006.

[4] J.W. Picone. Signal modeling techniques in speech recognition. *Proceedings of the IEEE*, 81(9):1215–1247, September 1993.

[5] T. Kinnunen and H. Li. An overview of text-independent speaker recognition: From features to supervectors. *Speech Communication*, 52(1):12 – 40, 2010.

[6] S. Stevens, J. Volkman, and E Newman. A scale for the measurement of the psychological magnitude of pitch. *Journal of the Acoustical Society of America*, 8(3):185–190, 1937.

[7] O. Pietquin, L. Couvreur, and P. Couvreur. Applied clustering for automatic speaker-based segmentation of audio materials. *Journal of Operations Research, Statistics and Computer Science (JORBEL)*, 41(1-2):1–12, 2001.

[8] M. Basseville. Distance measures for signal processing and pattern recognition. *Signal Processing*, 18(4):349–369, 1999.

[9] H. Frezza-Buet. Following non-stationary distributions by controlling the vector quantization accuracy of a growing neural gas network. *Neurocomputing*, 71(7-9), 2008.

[10] S. P. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[11] T. M. Martinez and K. J. Schulten. Topology representing networks. *Neural Networks*, 7(3):507–522, 1994.

[12] B. Fritzke. A self-organizing network that can follow non-stationary distributions. In *Proc. of ICANN'97*, pages 613–618. Springer, 1997.

[13] G. Chartrand, G. Kubicki, and M. Schultz. Graph similarity and distance in graphs. *Aequationes Mathematicae*, 55(1-2):129–145, 1998.

[14] L. F. Lamel, J.-L. Gauvain, and M. Eskénazi. Bref, a large vocabulary spoken corpus for french. In *Proc. of Eurospeech 91*, pages 505–508, 1991.