Unsupervised SVM based 2-Speaker Clustering

Binda Celestino, Hugo Cordeiro, Carlos Meneses Ribeiro

Multimedia and Machine Learning Group
Department of Electronic Telecommunication and Computer Engineering
Instituto Superior de Engenharia de Lisboa (ISEL), Portugal
celestinobinda@hotmail.com, {hcordeiro,cmeneses}@deetc.isel.ipl.pt

Abstract

This paper proposes two algorithms for the task of 2-speaker unsupervised clustering. The first one creates two SVM models, one for each speaker. The second creates only one SVM model, being each speaker assigned to each class of the same model. These clustering algorithms are based on traditional two-classes SVM and use MLSF coefficients as acoustic features to represent the speakers.

Tests were conducted in the audio stream of two interview videos in Portuguese, each one with two male speakers. Results must be considered as preliminary but if the speech segmentation was well conceived no errors were found.

Index Terms: speaker clustering, speech segmentation, speaker segmentation, support vector machine, mel line spectrum frequencies.

1. Introduction

There are several applications that demand the identification of speakers during a conversation. Typically, speaker segmentation and clustering is used as a pre-processing stage of another task, for example in automatic speaker recognition. The aim of speaker clustering and segmentation applied to a multi speaker conversation is to group all segments belonging to the same speaker. Normally, there is no *a priori* information of the number of speakers in the conversation as well as whom they are. Like a telephone conversation, whoever, only two speakers are involved and this can be known in advance.

Most common approaches to speaker segmentation use Bayesian Information Criterion (BIC) [1] or Generalised Likelihood Ratio (GLR) [2]. These are supported by Gaussian mixtures models (GMM). Alternative methods can use Support Vector Machine (SVM) [3] and multi-class SVM [4].

The more commonly used acoustic feature is the well known MFCC coefficients. Mel Line Spectrum Frequencies (MLSF) was also proposed in [5] as an alternative feature, that show to have similar performance to MFFC coefficients.

This paper proposes two algorithms for the task of 2-speakers unsupervised clustering. The first one creates two SVM models, one for each speaker. The second one creates only one SVM model, being each speaker assigned to each class of the same model. These clustering algorithms are based on traditional two-classes SVM and use MLSF coefficients as acoustic features to represent the speakers.

To have a complete speaker segment system, speech segmentation must be performed before speaker clustering. These speech segments must have speech only from just one speaker.

Tests were conducted in the audio stream of two interview videos in Portuguese, each one with two male speakers. Speaker segmentation is evaluated computing false alarm rate (FAR) and missed detection rate (MDR). Results must be considered as preliminary but if the speech segmentation was well conceived no errors were found.

The rest of the paper is organized as follows: the next section describes the proposed system to segment two speakers that include the proposed clustering algorithm; section 3 presents the experimental tests and results; and the last section concludes and launches some future work.

2. Proposed System

The proposed system for speaker segmentation is divided in two steps. The first step segments the input speech in segments where is more likely to contain speech from just one speaker. The second step clusters this speech segments and produces the final speaker segmentation. Clustering algorithm relies in SVM models trained with MLSF coefficients as acoustic features.

2.1 Speech segmentation

The first step of the speaker segmentation is to segment the input speech signal. This is achieved through an energy based voice activity detector (VAD) adapted from [6]. The main drawback of using an algorithm as simple as this emerges when speakers interrupt each other without any pause, creating a segment with two speakers. In consequence, a miss detection failure is introduced as the unsupervised clustering algorithm that is in the second step can not split these segments.

2.2 MLSF coefficients

MLSF was proposed in [5] as a feature that carries speaker information. This feature is a modification of the well known LSF coefficients to include perceptual information. This is achieved as long as the autocorrelation coefficients are computed as the inverse Fourier Transform of the mel-spectrum energies, originating mel-autocorrelation coefficients and therefore mel-line spectrum frequencies.

The encoding properties of the LSF are widely known and typically used in speech coders. MLSF coefficients have the same characteristics in the representation as the LSF, arranged on the unit circle, which can benefit from the advantages offered by the quantization to recognize speakers remotely without transmission of the speech signal itself.

Differences between the MLSF poles from the same frame also contains speaker information, as are related to the formants bandwidth.

2.3 SVM speaker model and frame score

MLSF coefficients are computed frame by frame and, for each speaker, a codebook of MLSF trained with the LBG algorithm acts as input feature.

A speaker is represented by a SVM model with a Gaussian kernel. This model is trained with the respective MLSF codebook against a world MLSF codebook (trained with several different speakers). All MLSF codebooks have the same number of codewords, independently from the length of the speech material. This accounts for different time lengths between speakers.

Evaluate a segment X in a model S_j trained for speaker j is frame based. Each frame is scored by the corresponding SVM model and the decision is made based on the frame score over the entire segment, defined by the rate of frames classified on the model:

$$K(X \mid S_j) = \frac{\text{number of frames classified in model } j}{\text{total number of frames}} \quad (1)$$

In the context of speaker identification, the higher frame score corresponds to the identified speaker. In the context of speaker verification, the frame score is compared to a given threshold. A speaker is accepted as true speaker if the frame score is higher than this threshold.

2.3 Unsupervised clustering algorithms

The two speakers clustering algorithm is based on [7], but with some major adaptations. Instead of based on LLRS over UBM with MFCC coefficients as features, this algorithm is based on SVM coefficients and use the frame score to measure the adaptation between segments and the speaker model, with MLSF as acoustic features. Also some simplifications are made to obtain the first approximation of the two speaker models. The algorithm is described as follows:

Stage 1: Initial models

1. Segment the test speech file.
2. Generate a speaker model S_1 based on the biggest segment, trained with a codebook of length 128 codewords, and fix one of the two speakers.
3. For all the segments with more than 2 seconds, compute the frame score. Generate a speaker model S_2 based on the lowest frame score, the more likely to be from the second speaker.

Stage 2: Initial clustering

4. All the remainder segments longer than 1 second were scored against speaker models S_1 and S_2. The difference between the respective frame scores are computed as:

$$\Delta s = K(X \mid S_1) - K(X \mid S_2) \quad (2)$$

5. The segment with biggest positive ΔS (more likely to be from speaker 1) is used to retrain model S_1.
6. The segment with lowest negative ΔS (more likely to be from speaker 2) is used to retrain model S_2.
7. Repeat steps 4 to 6 until no more segments longer than 1 second are left.

Stage 3: Final models

8. Use the speaker models S_1 and S_2 to compute the difference between frame scores ΔS in all segments longer than 1 second.
9. For the segments assigned to model S_1, create a new speaker model S_1 with the segments within the upper half ΔS. This new model is trained with codebooks of 1024 codewords.
10. For the segments assigned to model S_2, create a new speaker model S_2 with the segments within the bottom half ΔS. This new model is also trained with codebooks of 1024 codewords.

Stage 4: Final clustering

11. Use the speaker models S_1 and S_2 to compute ΔS for all the segments. If ΔS is positive the respective segment is assigned to speaker 1 and if ΔS is negative the respective segment is assigned to speaker 2.

The clustering algorithm is based in the difference of frame scores between the two speaker models. This suggests a new algorithm based on one single model, where the speakers are trained against each other. This new algorithm is described as follows:

Stage 1: Initial model

1. Segment the test speech file.
2. Generate a speaker model S_1 based on the biggest segment, trained with a codebook of length 128 codewords, and fix one of the two speakers.
3. For all the segments with more than 2 seconds, compute the frame score. Generate a two speaker's model S_{12} based on two segments: the biggest segment and the segment with the lowest frame score. The former is more likely to be from one speaker and the latter more likely to be from a second speaker.

Stage 2: Initial clustering

4. All the remainder segments longer than 1 second were scored against the two speaker's model S_{12}.
5. The segment with biggest frame score (more likely to be from speaker 1) and the lowest frame score (more likely to be from speaker 2) are used to retrain the two speakers model S_{12}, if they obtain more than 50% and less than 50% respectively.
6. Repeat steps 4 and 5 until no more segments longer than 1 second are left.

Stage 3: Final model

7. Use the two speaker's model S_{12} to compute the frame score in all segments greater than 1 second.
8. Create a new two speaker's model S_{12} based on top 25% frame scores (more likely to be from speaker 1) and the lowest 25% frame rates (more likely to be

from speaker 2). This new model is trained with codebooks of 1024 codewords.

9. If necessary for future identification or verification, create separate speaker models S_1 and S_2, based in the same assumption of step 8.

Stage 4: Final clustering

10. Use the two speaker's model S_{12} to compute the frame score for all the segments. If the frame score is bigger than 50% the respective segment is assigned to speaker 1, and if the frame score is less than 50% the respective segment is assigned to speaker 2.

If speaker identification of the two clustered speakers must be performed from a set of previous speaker models, the clusters segments take as a whole must be scored in this set of models and the higher frame score corresponds to the identified speaker.

3. Tests and Results

Tests were conducted in the audio stream of two interview videos in Portuguese, each one with two male speakers. For demonstration purpose, the identified speaker was subtitle in the original video.

As acoustic features, MLSF coefficients are computed frame by frame. Each frame has 20 ms and 50% overlap between frames. The vector feature order is 32, 16 MLSF coefficients and more 16 corresponding to temporal deltas. Like in cepstral mean subtraction, all features are normalized to have zero mean and unit variance.

For each speaker, a codebook of MLSF coefficients trained with the LBG algorithm acts as input feature to the SVM classifier. For the word model 10 male speakers from the "2002 NIST Speaker Recognition Evaluation Corpus" [8] are used.

The speaker SVM models are trained with a codebook of 128 codeword in the initial stages 1 and 2 of the algorithms and 1024 codewords in the final stages 3 and 4.

One of the two test files is well speech segmented, as the speakers do not interrupt each other. For this test file both the unsupervised clustering algorithms are able to segment the two speakers without any errors.

For the other test file the speech segmentation do not generate segments with only one speaker for all segments, as the speakers tends to interrupt each other. As the clustering algorithms are not able to split these segments, detection failures are introduced. However, the segments containing only one speaker are well grouped.

In order to check if the cluster algorithms perform well if the segmentation was well done, the segments with two speakers was hand segmented before running the cluster algorithms. Without surprise no errors are found in the final speaker segmentation.

4. Conclusions and future work

This work presents an unsupervised clustering algorithm of two unknown speakers. The algorithm is based on SVM and uses the frame score to measure the adaptation between segments and the speaker model.

The feature adopted is a codebook of MLSF coefficients. MLSF coefficients have the same characteristics than LSF, benefit from the advantages offered by the quantization to recognize speakers remotely without transmission of the speech signal itself.

Before clustering, segmentation of the input speech must be performed, in segments containing speech from just one speaker. The proposed speech segmentation, based in energy, can not discriminate speakers interrupting each other, compromising the clustering. However, if speech segmentation is well conceived, preliminary tests shows the feasibility of the proposed unsupervised clustering algorithm. Although tested with only two files, no errors are found.

A set of improvements can be performed in this system. This will focus future work in several directions, namely:

(1) Improve the speech segmentation process, particularly when the speakers are interrupting each other. This implies to discard a simple VAD based in energy speech segmentation and adopt a more complex speaker segmentation method, which finds speaker changes based on maxima of some distance measure between two adjacent windows shifted along the speech signal.

(2) Expand the clustering algorithm to detect multi-speakers. This can be obtained with a multi-class SVM model;

(3) Identify the clustered speakers if they belong to a set of speakers with known models. This is performed evaluating all the models. The model with the higher frame score corresponds to the identified speaker.

Finally, evaluation with a more extensive and consistent corpus must be performed and results must be compared with a reference system.

5. References

[1] Rissanen, J., "Stochastic Complexity in Statistical Inquiry. Series", Computer Science, 1989, Vol. 15. World Scientific, Singapore, Chapter 3, 1989

[2] Gish, H., Siu, M.-H., Rohlicek, R., "Segregation of speakers for speech recognition and speaker identification." IEEE International Conference on Acoustics Speech and Signal Processing, 1991. 873-876, 1991

[3] Fergani B., Davy M., Houacine A., "Speaker Diarization using one-class support vector machine" Speech Communication 50, 355-365, 2008

[4] Nazari, M., Faez K., "Speaker Detection and Clustering with SVM Tecnique in Persian Conversational Speech", SETIT 2007, Sciences of Electronic, Technologies of Information and Telecommunications , 2007

[5] Cordeiro, H., Meneses Ribeiro, C., "Speaker characterization with MLSFs", *IEEE Odyssey 2006, the Speaker and Language Recognition Workshop,* Porto Rico, 2006.

[6] Lamel, L., Rabiner, L., Rosenberg A., Wilpon, J., "An Improved Endpoint Detector for Isolated Word Recognition", *IEEE Transactions on Acoustics, Speech, and Signal Processing,* volume ASSP-29, Nº 4, 777-785, 1981.

[7] Deng, J., Zheng, T. F., and Wu, W., "UBM Based Speaker Segmentation and Clustering for 2-Speaker Detection", ICSLP 2006, 116-125, 2006.

[8] "2002 NIST Speaker Recognition Evaluation Corpus", Online: http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2004S04