KNNDIST: A Non-Parametric Distance Measure for Speaker Segmentation

Seyed Hamidreza Mohammadi[*1], *Hossein Sameti*[2],
Mahsa Sadat Elyasi Langarani[2], *Amirhossein Tavanaei*[2]

[1] Center for Spoken Language Understanding, Oregon Health & Science University, Portland, OR
[2] Speech Processing Laboratory, Sharif University of Technology, Tehran, Iran
mohammah@ohsu.edu, sameti@sharif.edu, {mselyasi, tavanaei}@ce.sharif.edu,

Abstract

A novel distance measure for distance-based speaker segmentation is proposed. This distance measure is non-parametric, in contrast to common distance measures used in speaker segmentation systems, which often assume a Gaussian distribution when measuring the distance between two audio segments. This distance measure is essentially a k-nearest-neighbor distance measure. Non-vowel segment removal in pre-processing stage is also proposed. Speaker segmentation performance is tested on artificially created conversations from the TIMIT database and two AMI conversations. For short window lengths, Missed Detection Rated is decreased significantly. For moderate window lengths, a decrease in both Missed Detection and False Alarm Rates occur. The computational cost of the distance measure is high for long window lengths.

Index Terms: speaker segmentation, distance measure, k-nearest-neighbor

1. Introduction

The aim of speaker segmentation is to extract the longest possible homogenous segments in a conversation. A homogenous segment is considered to be generated from a single source (speaker). In a segmentation problem, it is assumed that no prior knowledge is available about the speakers (no speaker models) [1].

Speaker segmentation is the first essential part in speaker diarization systems. Speaker diarization systems are used in speaker tracking and rich transcription systems [2]. Such systems have been applied to several real-world databases such as Broadcast News data [3] and Rich Transcription of Meetings [2].

Speaker segmentation algorithms are divided into two broad categories: distance-based and model-based. The model-based algorithms assume a priori knowledge about the speakers (it has a speaker model training phase). In contrast, the distance-based segmentation detects acoustic changing points in an unsupervised manner. The distance-based speaker segmentation uses a distance measure to compute a distance curve. The speaker changing points are then extracted by processing the distance curve. The distance is usually derived from a statistical framework [4]. In other words, it is assumed that feature vectors in two audio segments are generated from a certain probability distribution (e.g. the multivariate Gaussian). The distance measure between the two audio segments is computed by computing the dissimilarity of these two distributions [4]. There are a number of distance measures that have been widely used. The KL2 distance measure is proposed in [5]. It uses Relative Cross Entropy or Kullback–Leibler distance to compute the distance between the Gaussian distributions. The Generalized Likelihood Ratio (GLR) is also proposed [2, 6] which assumes two hypotheses, and computes the distance by computing the likelihood of each hypothesis. One of the most prominent distances used is the Bayesian Information Criterion (BIC) which has been studied in [7]. Several other improvements over BIC have been proposed. XBIC is proposed in [8]. It uses the cross-probabilities between each audio segment and the model trained with data from the other segment. T^2BIC is also proposed in [9] which haw less computational cost. It takes into account the Hotelling's T^2-Statisics to pre-segment the audio quickly. A post-processing step is then followed to decide to accept or decline the selected changing points.

The distance measures used so far are mostly parametric methods. As mentioned before, these measures usually assume that feature vectors are generated from a certain probability distribution, such as multivariate Gaussian distribution. When using short to moderate length windows (as you will see later in the paper) the modeling is usually not accurate enough. In this study, a non-parametric distance measure is proposed. This distance measure is based on the k-nearest-neighbor distance measure. In contrast to previous distance measures, it does not assume any certain probability distribution.

In Section 2, the speaker segmentation algorithm is explained. The BIC distance measure and the proposed distance measure are described in Section 3. In Section 4, the database that is used to evaluate the algorithm and also the evaluation results are described. The conclusion of this study is presented in Section 5. Some guidelines for future work are also described in this section.

2. Speaker Segmentation

2.1. Signal Processing

The features used in speaker segmentation systems are usually the same features that are used in speaker recognition systems. In this study, Mel-Frequency Cepstral Coefficients (MFCCs) are used as features. Prior to feature extraction, the input waveform is pre-emphasized (with coefficient of 0.97) and framed (25ms frame size and 10ms frame shift). Then, a Hamming window is applied to each frame. For each frame, 12 MFCCs are extracted (using 40 Mel-scaled filters). The energy of each frame is also appended to the feature vector.

[*] The author *Seyed Hamidreza Mohammadi* worked on this study as part of his Master thesis in Sharif University of Technology.

2.2. Non-Vowel Removal

In speaker segmentation algorithms, non-speech parts of conversation are removed and the processing is done on speech parts. In this study, it is proposed that in addition to non-speech parts, non-vowel segments should also be removed. This is because vowel regions are more important for speaker distinction in short windows. The results showed that this increases the performance of the system, especially the False Alarm Rate as will be shown Section 4.

As mentioned before, silence frames are removed prior to applying the segmentation algorithm. Two Gaussian Mixture Models (GMMs) are trained on vowel and non-vowel frames. The number of components is chosen to be 128 components. These two models are used to classify the incoming frames based on whichever has higher likelihood. MFCC and Zero Crossing Rate (ZCR) are used for this modeling vowel and non-vowels, since they are good feature for discriminating between vowels and non-vowels. The frames are chosen from TIMIT database. The phonemes are excluded from the rest of the testing data that is used for evaluation.

2.3. Speaker Segmentation Algorithm

In this study, fixed-sliding-window algorithm is used for detecting speaker changing points. As the name suggests, a fixed-size window is slid over the whole conversation, resulting in a distance curve. By processing this distance curve, speaker changing points are detected. These detected changing points will be compared to the reference changing points to evaluate the performance of the system. This process is shown in Fig. 1.

Given a distance curve, a "significant" local maximum which has a positive ΔBIC value is considered to be a speaker changing point [10].

2.4. Evaluating Speaker Segmentation Algorithms

For the purpose of evaluating speaker segmentation algorithms, we have used False Alarm Rate (FAR) and Missed Detection Rate (MDR) [10]. These two measures are defined as in (2).

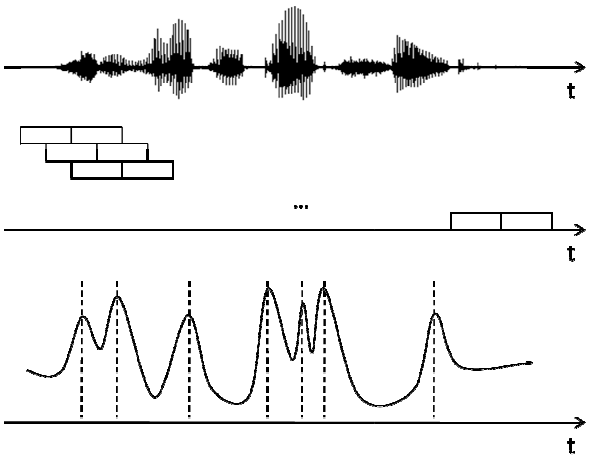

Figure 1: *Fixed –sliding-window speaker segmentation*

$$FAR = 100 \times \frac{FA}{RC + FA}\%, MDR = 100 \times \frac{MD}{RC}\% \quad (2)$$

The terms FA, MD and RC in (2) represent the number of false detected changing points, the number of missed changing points and the number of true changing points, respectively. For every reference changing point, if a changing point is not detected within a span of 0.5 seconds, it will be considered a missed detection. For a reference changing point, if more than one changing point is detected within the 0.5 seconds span, all the additional detected changing points are considered false alarm. For every detected changing point, if a reference changing point is not present within a span of 0.5 seconds, it will be considered a false alarm.

3. Distance Measure

3.1. Bayesian Information Criterion

As described before, the most essential part of the distance-based speaker segmentation and clustering systems is the distance measure. The distance measure is used to compare the similarity of two audio segments. Usually, in speaker segmentation systems the distance is computed between two consecutive audio segments. These audio segments are called "windows" in most of the speaker segmentation studies.

The BIC distance measure assumes one of the following two hypotheses is true for a given window. Either a window is generated from a single source or two different sources [10]:

- $H_0 : (x_1, ..., x_{N_Z}) \sim N(\mu_Z, \Sigma_Z)$: The window is uttered by one single speaker; hence one Gaussian is adequate for modeling the window.
- $H_1 : (x_1, ..., x_{N_X}) \sim N(\mu_X, \Sigma_X) and (x_1, ..., x_{N_Y}) \sim N(\mu_Y, \Sigma_Y)$ The window is uttered by two different speakers; hence two Gaussians should be used to model the window.

The BIC for each hypothesis is computed as in (3). The term $log\ (X, H)$ represents the maximum likelihood criterion and the term d represents the dimension of the data [7]. The second part of the equation is a penalty term, which penalizes model complexity.

$$BIC(H) = \log(X, H) - \lambda \frac{d}{2} \log N_X \quad (3)$$

The BIC value of the first hypothesis is subtracted from the BIC value of the second hypothesis, giving the final distance measure called the ΔBIC [10]. Assuming the distributions to be Gaussian, this value is computed using (4). The terms N_X, N_Y and N_Z represent length of windows X, Y and Z, respectively. Whenever this value is positive, the window is considered to be a changing point and whenever this value is negative, the window is uttered by a single speaker. This value can also be used as a distance measure between two audio segments. It is essentially GLR distance measure minus a penalty term. The GLR part of the ΔBIC formula measures the dissimilarity between the Gaussian distributions.

$$\Delta BIC = \frac{N_Z}{2}\log|\Sigma_Z| - \frac{N_X}{2}\log|\Sigma_X| - \frac{N_Y}{2}\log|\Sigma_Y| - \frac{\lambda}{2}\left(d + \frac{1}{2}d(d+1)\right) \quad (4)$$

3.2. K-Nearest-Neighbor Distance Measure

Most of distance measures used in speaker segmentation and clustering assume that feature vectors are generated from a Gaussian probability distribution. The distance is then computed by a dissimilarity measurement between distributions of two audio segments.

In this study, a k-nearest-neighbor distance measure is proposed as distance measure. The distance measure between segments X and Y is computed as follows:

Normalize dimensions of the features to the [0, 1] range.

Apply an appropriate weight to each dimension according to relevance of that dimension (in this study, uniform weights are used).

Find *k* pair of frames between two audio segments which have the lowest Euclidian distance.

The final distance is computed by summing the distances between all these k pair of frames.

The above procedure is formulated as (4). The function $\min_k$ returns the *k* lowest values of all distances.

$$KNNDIST(X,Y,k) = \sum \min_k \left(euclid(x_m, y_n)\right), m \in M, n \in N \quad (4)$$

3.3. KNNDIST versus BIC

The idea behind the proposed distance measure is depicted in Fig. 3. For simplicity, two dimensional data are used. Each sample is a data-point in each of the windows. During the computation, five nearest pairs are selected. The final distance is the sum of the distances between each of these five pairs. This is in contrast to other distances (such as KL2) which assumes a Gaussian probability distribution for each window and then compare these distributions. This idea is depicted in Fig. 4. As you can see, each window's data-points are modeled by a Gaussian distribution. The final distance is computed using the parameters of the model.

Two sample distance curves of a speech segment are shown in Fig 5. The above figure is the distance curve extracted using BIC distance measure. The below figure is the distance curve extracted using KNNDIST distance measure. The window length is set to 2 seconds. The speaker changing points result in more sudden peaks in KNNDIST distance curve compared to BIC distance curve. Also, the peaks are right on the changing points, unlike the BIC distance curve, where the peak may occur further away from the actual changing point (because of the parametric characteristic of BIC). Also, it seems that fewer fluctuations occur in a homogenous speaker segment, resulting in less False Alarm Rate.

The computational complexity of computing BIC distance is bound to complexity of computing determinant of the covariance matrix. If we let n represent the size of the window and d represent the dimension, the computational complexity is $O(d^2n + d^{2.8})$, which is equal to $O(d^2n)$ for $n \gg d$ (which is usually the case). The computational cost of computing KNNDIST is $O(dn^2+n^2\log n^2)$. Because of the n^2 factor, KNNDIST is more suited for small window sizes.

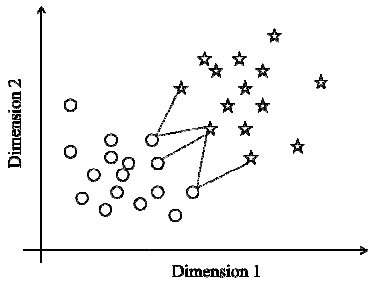

Figure 3: computing neighboring samples in *KNNDIST distance measure (for k=5)*

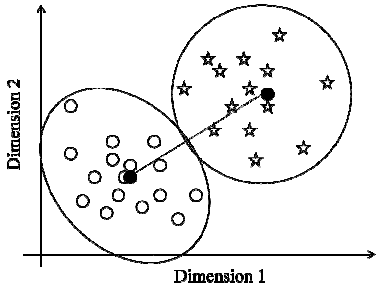

Figure 4: *Gaussian assumption over audio segments*

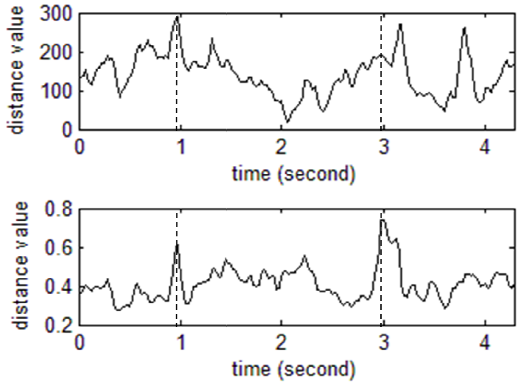

Figure 5: *BIC sample distance curve (above) KNNDIST sample distance curve (below)*

4. Results and Discussion

For the purpose of evaluating the proposed distance measure, two sets of conversations are artificially created by concatenating speech from TIMIT database. In the first one (conversation set A), there are 4389 speaker turns in all these conversations (each speaker segment is 1.5 to 4 seconds). The number of speakers in one conversation varies from 2 to 6. In the second one (conversation set B) only 10 seconds segments exist. There are 835 speaker turns in this set. Also two conversations from AMI multimedia meetings database is also used to validate the newly proposed distance and also the non-vowel removal idea in real environments.

Three experiments were performed on these two datasets. The first one shows the performance of the proposed distance measure with different window lengths on conversation set A. In

the second one, the DET curve for system performance on set A for both BIC and KNNDIST distance measure is plotted. In the third one, system accuracy is tested on the conversation set B. The features used are 12 MFCCs appended by log-energy. In all of the experiments, the value of k is set to half of the size of the window length (e.g. if a window consists of 200 frames, k is set to 100).

In the first experiment, the system is tested using different window lengths. The window length is set to 0.5, 0.75, 1, 1.5, 2 and 3 seconds. The γ in (1) is set to 0.2 for this experiment. The results are shown in Table 1. As it is evident, for short window lengths, the improvement in Missed Detection Rate is very high (about 70% decrease in MDR, with almost the same FAR). In the second experiment, window size 2 seconds with γ value 0.2 is tested on conversation set B. Both MDR and FAR are far less compared to BIC distance measure. Also the results on AMI database show a significant decrease in FAR while the MDR stays almost the same.

The DET curves of the segmentation system accuracy using both distance measures are shown in Fig. 6. For plotting the curve, the γ value in (1) is changed from 0.05 to 2 with a 0.1 step. In this experiment the window length is set to 1.5 seconds. The system is tested on set A. It can be seen from the DET curve that KNNDIST distance results in a far better performance compared to BIC distance measure.

The effects of removing non-vowel segments are also studied. We evaluate the following configuration: BIC distance measure with 2 seconds window size on set A. The results for only silence removal are MDR of 8.13% and FAR of 36.17%. For non-vowel removal, the results are MDR of 8.15% and FAR of 20.80%.

5. Conclusion and Future Work

A non-parametric distance measure for speaker segmentation was proposed in this study. It was shown that by using this distance measure, the performance of the speaker segmentation system is increased. For short window length (< 1.5 seconds), the increase of performance was in the Missed Detection Rate (about 70% decrease in MDR). For moderate window lengths (about 2 seconds), there is a slight increase in performance in both FAR and MDR. For long window lengths (> 2.5 seconds), the performance is decreased. Commonly, moderate window length (about 2 seconds) is used in segmentation systems. The computational complexity of this distance is $O(n^2d)$, so it is better suited to be used with short and moderate window lengths.

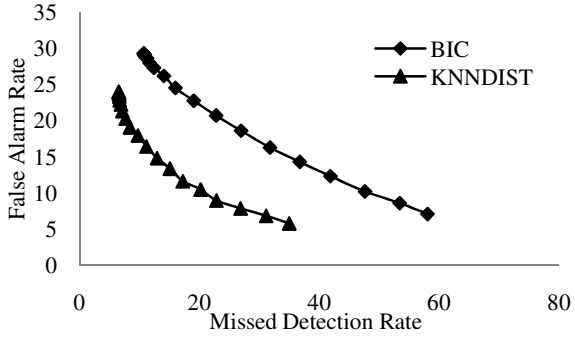

Figure 6: *DET curves for BIC and KNNDIST*

Table 1. Results for different window lengths

Window Length (seconds)	Database	Results			
		BIC		KNNDIST	
		MDR	*FAR*	*MDR*	*FAR*
0.5	A	29.91%	44.53%	11.57%	43.57%
0.75	A	23.35%	37.67%	8.15%	37.88%
1	A	14.78%	33.05%	5.80%	34.27%
1.5	A	10.73%	29.25%	6.53%	25.29%
2	A	8.15%	20.80%	7.24%	17.21%
3	A	13.01%	14.42%	14.14%	17.56%
2	B	44.31%	61.21%	25.86%	51.62%
2	AMI IS1008a	26.13%	48.84%	26.19%	37.33%
2	AMI ES2008b	29.72%	53.98%	28.83%	39.69%

In this study, the number of k neighbors was selected to be the half of the window length. No experiment was performed to find an optimal k value. A study about finding an optimal value for k can be performed.

In this study, all of the data dimensions were assigned the same value (weight) in computing the KNNDIST distance measure. A study can be devoted to finding an optimal weight set in computing the distance.

6. References

[1] Kotti, M., Moschou, V. and Kotropoulos, C., "Speaker segmentation and clustering", Signal Processing, Volume88, Issue 5, 2008, pp. 1091-1124.

[2] Fiscus, J. G., Ajot, J. and Garofolo, J. S., "The Rich Transcription 2007 Meeting Recognition Evaluation", Multimodal Technologies for Perception of Humans, Springer-Verlag Berlin, 2008.

[3] Barras, C., Zhu, X., Meignier, S., and Gauvain, J. L., "Multistage speaker diarization of broadcast news", IEEE Trans. on Audio, Speech and Lang. Proc.. 14 (5), 2006, pp. 1505–1512.

[4] Cheng, S., Wang, H. and Fu, H., "BIC-Based Speaker Segmentation Using Divide-and-Conquer Strategies with Application to Speaker Diarization", IEEE Trans. on Audio, Speech, and Language Processing, volume 18, number 1, pages 141-157, January 2010.

[5] Siegler, M., Jain, U., Raj, B. and Stern, R., "Automatic segmentation, classification and clustering of broadcast news audio," in Proc. DARPA Speech Recognition Workshop, Feb. 1997, pp. 97–99.

[6] Gish, H., Siu, M.H. and Rohlicek, R., "Segregation of speakers for speech recognition and speaker identification", in: Proceedings of the 1991 IEEE International Conference on Acoustics, Speech, and Signal Processing, Toronto, Canada, April 1991, pp. 873–876.

[7] Chen, S.S. and Gopalakrishnan, P. S., "Speaker, environment and channel change detection and clustering via the Bayesian. Information Criterion", DARPA Speech Recog. Workshop, 1998.

[8] Anguera, X.,, "XBIC: Real-time cross probabilities measure for speaker segmentation", Technical Report TR-99-2004, ICSI, 2005.

[9] Zhou, B. and Hansen, J. H. L., "Efficient audio stream segmentation via the combined T2 statistic and the Bayesian information criterion", IEEE Trans. Speech Audio Process. 13 (4) (July 2005) 467–474.

[10] Delacourt, P. and Wellekens, C. J., "DISTBIC: a speaker-based segmentation for audio data indexing", Speech Communication.32, 2000, pp. 111–126.