Experiments with Segmentation in an Online Speaker Diarization System

Marie Kunešová[1,2(✉)], Zbyněk Zajíc[1], and Vlasta Radová[1,2]

[1] NTIS - New Technologies for the Information Society, Faculty of Applied Sciences, University of West Bohemia, Univerzitní 8, 306 14 Plzeň, Czech Republic
{mkunes,radova}@kky.zcu.cz, zzajic@ntis.zcu.cz
[2] Department of Cybernetics, Faculty of Applied Sciences, University of West Bohemia, Univerzitní 8, 306 14 Plzeň, Czech Republic

Abstract. In offline speaker diarization systems, particularly those aimed at telephone speech, the accuracy of the initial segmentation of a conversation is often a secondary concern. Imprecise segment boundaries are typically corrected during resegmentation, which is performed as the final step of the diarization process. However, such resegmentation is generally not possible in online systems, where past decisions are usually unchangeable. In such situations, correct segmentation becomes critical. In this paper, we evaluate several different segmentation approaches in the context of online diarization by comparing the overall performance of an i-vector-based diarization system set to operate in a sequential manner.

Keywords: Speaker diarization · Speaker change detection · i-vectors · Convolutional neural network

1 Introduction

Speaker diarization is a speech processing task which aims at categorizing different speech sources in a conversation of two or more speakers, such that utterances produced by the same speaker are assigned the same label. In other words, we are trying to determine "Who speaks when?", typically without any prior knowledge about the number and identities of the speakers.

Speaker diarization systems can be divided into two main categories: *offline* and *online*. Offline systems process a given audio recording as a whole, requiring that the entirety of the data is available at the beginning of the process. This allows these systems to use all available information for their decisions. *Online* systems, by contrast, operate in a strict left-to-right manner and can process an incoming audio stream in real-time. The decisions made by these systems can be based only on previously seen data, independent of future information, and once made, cannot be changed.

The most common diarization approach, used by both offline and online systems, consists of two main steps: segmentation and clustering. The input signal

K. Ekštein and V. Matoušek (Eds.): TSD 2017, LNAI 10415, pp. 429–437, 2017.
DOI: 10.1007/978-3-319-64206-2_48

is split into short intervals and these are then merged into clusters corresponding to the individual speakers. Common algorithms include clustering based on the Bayesian information criterion (BIC) [11] or on distances between i-vectors [12]. In the case of offline systems, there is often an additional resegmentation step, which refines the original segment boundaries.

An alternative diarization approach combines segmentation and clustering into a single iterative process, often with the use of Hidden Markov Models (HMMs) or related concepts [1,9]. However, this approach is not typically used in online diarization.

There are many possible ways to perform segmentation. Ideally, we want to have segments which only contain a single speaker. This is best achieved with speaker change detection (SCD) - identifying possible speaker boundaries and then splitting the conversation there. Common approaches include the Bayesian Information Criterion (BIC), Generalized Likelihood Ratio (GLR) [11], Support Vector Machines (SVM) [5] or Deep Neural Networks (DNNs) [7,15].

However, the SCD approach is problematic in spontaneous telephone conversations. These typically contain very short speaker turns and frequent overlapping speech, which makes it difficult to correctly detect speaker turns. For this reason, most authors in the telephone domain (e.g. [6,12,13]) choose to simply cut the conversation into very short intervals of fixed length. It is assumed that any inaccuracies can be resolved during the later stages of the diarization process, typically by performing a final resegmentation step. This was also confirmed in our recent paper [16], which compared the fixed length approach with SCD-based segmentation using GLR distance. There, we showed that even though the SCD approach led to better initial clusters, the final results of both options after resegmentation were comparable.

Unfortunately, no such resegmentation is possible in online systems. This means that a proper initial segmentation again becomes important.

This paper is in part inspired by the recent work of Zhu and Pelecanos [19], who have proposed an incremental adaptation process for online i-vector based speaker diarization. Their original paper focuses mainly on the clustering step and sidesteps the question of segmentation by utilizing *oracle* segmentation based on reference transcripts (although this has very recently been extended with ASR-based segmentation [3]). In our work, we follow up on their results by implementing the suggested online approach in our own diarization system, while using a different segmentation.

The main goal of this paper is then to compare multiple different segmentation options in the context of online speaker diarization. For this purpose, we use the aforementioned i-vector-based system and evaluate its performance on telephone data from the CALLHOME American English corpus [2]. As most telephone conversations involve only two individuals, our system explicitly assumes the presence of only two speakers and we limit our experiments to the two-speaker subset of the corpus.

2 Offline Diarization System

For our online diarization experiments, we have re-purposed an originally *offline* state-of-the-art diarization system which is based on i-vectors. In this section, we describe this base offline system, while the subsequent adjustments to a more sequential approach will be presented in Sect. 3.

The basic structure of the system is based on the i-vector approach which has recently become standard in speaker diarization [12,19]. The specific implementation largely follows the descriptions presented in our previous papers [8,16] and a diagram of the main steps can be seen in Fig. 1. The diarization process starts with the extraction of acoustic features from the conversation, followed by its splitting into short segments. This segmentation step can use one of multiple possible approaches, some of which will be explored in Sect. 4.

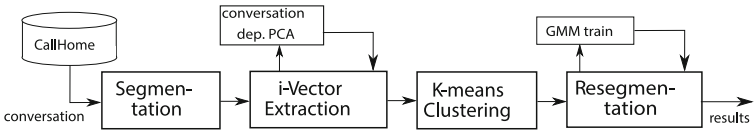

Fig. 1. Diagram of the offline diarization system.

As the next step of the process, we obtain a simplified representation of the individual segments. For each segment of the conversation, we first accumulate a supervector of statistics [17], from which we subsequently extract an i-vector via Factor Analysis [4]. The size of the i-vectors is further reduced with the aid of a conversation-dependent Principal Component Analysis (PCA) transformation [14].

Following this, the i-vectors are clustered in order to determine which parts of the conversation were produced by the same speaker. As we limit our data to conversations between only two speakers, we can use a simple k-means algorithm based on cosine distance between i-vectors [14].

Finally, we perform a frame-wise iterative resegmentation based on Gaussian Mixture Models (GMMs) trained on the data from each cluster. This serves to refine the speaker boundaries and correct mistakes caused by imprecise segmentation.

3 Online System

As our main goal was simply to investigate the sequential segmentation and clustering process, without the need for actual real-time output, we decided against implementing a complete, fully online diarization system. Rather, we have simply adjusted the original offline process which was described in Sect. 2 so that each of the steps separately operates in a left-to-right manner. As such, the initial steps of both systems are identical. However, the original k-means

clustering is replaced by a sequential algorithm, while both the conversation-dependent PCA reduction of i-vectors and the final resegmentation step, which are not possible to perform online, are removed entirely.

As the clustering step, we employ the i-vector adaptation process proposed by Zhu and Pelecanos [19], which is given by

$$T_n = \alpha V_n V_n^T + (1 - \alpha_n) I\,, \qquad \alpha_n = \frac{n}{n+R}\,, \tag{1}$$

where n is the number of i-vectors which have been processed so far, V_n is the first principal component of the i-vectors, T_n is an i-vector transformation matrix and R is the relevance factor which controls the rate of the adaptation.

The resulting sequential clustering then works as follows: For each new i-vector (which corresponds to a new segment), we first update the transformation matrix T_n using the formula in (1) and use it to transform all i-vectors seen up to this point. Then we calculate the cosine distance between the new transformed i-vector and all existing clusters, where the distance to a cluster is calculated as the average of the distances to all of its i-vectors. If the distance to the closest cluster is lower than a threshold (we will designate this threshold as θ) or the maximum number of clusters is reached (in our case, this number is 2), the new i-vector is assigned to this cluster. Otherwise, a new cluster is created.

Because all decisions made by the system are final and unchangeable, an incorrect decision at an early point in a recording can significantly impact the rest of the clustering process. In this regard, extremely short segments, particularly those under 0.5 s are the most problematic, as they typically do not contain sufficient information about the speaker in order to be correctly clustered.

As some of the segmentation approaches which we compared may produce such short segments, it was necessary to slightly adjust the clustering algorithm in order to avoid this issue. We achieve this by excluding any segments under 1 s in length from the regular clustering process. Instead, the corresponding i-vectors (which we do not consider to be representative of any speakers) are simply labeled as the nearest existing cluster (they are never used to create a new one), but they are not included in the calculation of T_n in (1) and we also do not consider them in later distance calculations.

4 Segmentation

In this paper, we compare several different segmentation approaches. For these experiments, we chose to use the segmentation algorithms which were previously described in our two recent papers [8,16] in the context of *offline* speaker diarization. All of these segmentation approaches assume the possibility of their use in online diarization, i.e. they operate sequentially or can be relatively easily adjusted in such manner.

Some of the described approaches rely on information about the presence of silence and speech which would under real conditions be provided by a voice activity detector (VAD). However, in order to avoid any specific VAD method

from influencing the results of the segmentation, we chose to use *oracle* VAD obtained from the reference transcripts and we discuss the possible dependence on VAD in the description of each segmentation method.

When performing segmentation, it is also important to consider the length of the resulting segments. In particular, we need to have a sufficient amount of information in order to be able to extract an i-vector from each segment. Typically, a segment length of at least 1–2 s is considered to be the minimum in order to obtain i-vectors which are representative of the speakers.

4.1 Fixed Length Segments

The simplest segmentation option is to split the input stream into short intervals of equal length, without considering any potential speaker boundaries. In our system, we follow the example of [12] by using overlapping segments. This allows us to increase the amount of information contained in a single i-vector while retaining a higher precision of the segmentation. Specifically, we chose to use segment length of 2 s with a 1 s overlap between neighboring segments.

4.2 GLR-Based Speaker Change Detection

As the first speaker change detection approach, we used the Generalized Likelihood Ratio (GLR)-based algorithm described in our previous paper [16]. This is a two-pass algorithm, which means that it is not suitable for true online diarization in its current form. However, we believe that it should be possible to implement a relatively similar algorithm in a strictly left-to-right form.

In the original two pass approach, which we used here, the algorithm first calculates the GLR distance between two sliding windows over the entirety of the conversation. A smaller number of the most likely speaker change points are found as the local maxima whose topographical prominence exceeds a set threshold. During the second pass, segments above a specific length are split according to the algorithm suggested in [16].

Finally, any segments which contain only a small percentage of speech frames (as determined by VAD), are labeled as silence and subsequently discarded. This means that the performance of this approach depends on VAD implementation and also gives it an advantage over the other approaches when reference VAD is used (as was done in our experiments).

4.3 CNN-Based Speaker Change Detection

The second SCD-based segmentation method which we considered uses a Convolutional Neural Network (CNN) as a regressor. We employ a CNN which was trained on spectrograms of acoustic signal using the method described in [8]. The reference training labels were in the form of a fuzzy labeling function L. Figure 2 depicts an example of a spectrogram, the values of the labeling L and the CNN output as a probability of speaker change P. Speaker changes are then identified

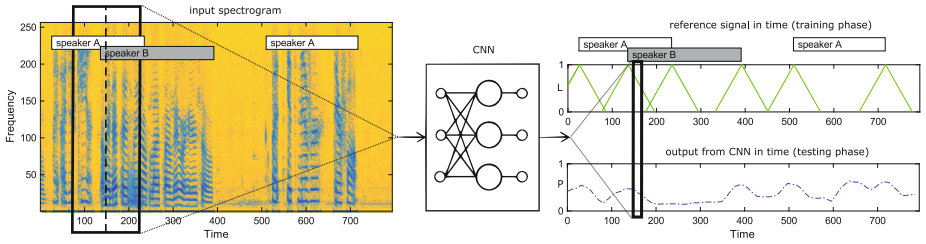

Fig. 2. The input speech as spectrogram is processed by the CNN into the output function P (a probability of change in time). The L-function (the reference speaker change) for the CNN training is depicted on top.

as peaks in the signal P using non-maximum suppression with a window size of 5 samples (0.5 s). We also apply a threshold of 0.5 on the detected peaks in order to remove insignificant local maxima. The signal between two detected speaker changes is considered as one segment.

Additionally, we also utilize the information about the change P from the CNN for weighting of the acoustic data in a segment, in order to refine the statistics accumulation process used for the subsequent i-vector generation [18].

In the offline version of this segmentation approach, we discard any segments under 1 s in length, as they are considered unreliable. They are only processed later during resegmentation. However, in the online variant, which does not have resegmentation, we keep all segments, regardless of length.

Processing the spectrogram window using a CNN takes only a very short time, which makes this approach suitable for online diarization. It should also be noted that the network is trained to detect all types of speaker boundaries and as such, does not need any information from a voice activity detector.

4.4 Oracle Segmentation

For comparison purposes, we also implemented oracle segmentation. In this approach, the conversations are split according to the reference transcripts: each individual record from the transcript becomes a single segment. As many of these segments are very short (often under 1 s), we adjust them slightly by joining any two segments from the same speaker which are separated by a silence of less than 0.5 s (this does not, however, eliminate all short segments). Otherwise, the segments are kept exactly as recorded in the transcripts, including any partial overlaps.

5 Results

For the evaluation of our system, we used the CALLHOME American English corpus of telephone speech [2], with both channels mixed into a single one. As 35 of the recorded conversations had been used for training the CNN which we

use for one of the segmentation approaches, we limited our experiments to the remaining 77 conversations with only two participants.

The results are evaluated using the Diarization Error Rate (DER), as defined by NIST [10], with the customary tolerance collar of 0.25 s around speaker boundaries. Contrary to a common practice in telephone speech diarization, we do not ignore overlapping segments during the evaluation. However, our listed error rates only include two of the three components of DER: missed speech (speech incorrectly labeled as silence) and speaker error (speech labeled as the wrong speaker). False alarm (silence incorrectly labeled as speech) is removed before evaluation with the help of the reference transcripts.

In Table 1, we present the results achieved with the four segmentation methods for a fixed decision threshold $\theta = 0.6$ and different values of the relevance factor R, which controls the rate of the adaptation (see Sect. 3). This includes $R = \infty$, which is equal to not using adaptation. We may notice that the adaptation process proposed in [19] can improve the final DER in all four cases, but the individual segmentation approaches have different optimal values of R.

For comparison, we also show the results of the offline system (adapted from our previous works [8,18]).

Table 1. Offline and online diarization results for different segmentation approaches, measured in terms of DER [%]. R is the relevance factor of the i-vector adaptation, with the value of ∞ being equal to no adaptation. Decision threshold for the online approach was $\theta = 0.6$. Offline results (except oracle) were adapted from [8,18].

	Offline	Online								
R	–	∞	8192	...	1024	512	256	128	64	32
Fixed length	9.23	18.62	**18.47**	...	18.88	19.34	19.80	20.43	–	–
GLR	11.98	15.04	–		14.29	14.15	14.12	**13.74**	14.23	14.29
CNN	7.84	15.16	–		**14.77**	14.95	14.91	15.93	–	–
oracle	6.80	10.98	–		–	9.60	9.58	**9.30**	9.78	10.71

The table shows that in the offline scenario, the naïve fixed length segmentation produces reasonable results (likely due to resegmentation [16]), although it is surpassed by the CNN-based approach. However, in our online system, this simple option is no longer sufficient, achieving nearly double the error of the oracle option. This suggests that correct segmentation is much more important in online systems.

Of the two SCD-based approaches, the GLR-based method scored better. However, this may be influenced by its reliance on VAD (as discussed in Sect. 4.2). As our experiments used oracle VAD from reference transcripts, this gives the approach an advantage compared to the CNN-based option, which did not use any information from VAD.

6 Conclusion

In this paper, we compared several different segmentation approaches in an i-vector-based speaker diarization system operating in a left-to-right manner. We have found that the final system performance highly depends on the quality of the segmentation step. In particular, the simple naïve splitting by fixed length, which is commonly used in offline systems, does not appear to be sufficient for an online approach. Instead, more sophisticated methods are required, such as one of the other approaches which we explored here.

Acknowledgments. This research was supported by the Ministry of Culture of the Czech Republic, project No. DG16P02B009.

References

1. Bozonnet, S., Evans, N.W., Fredouille, C.: The LIA-EURECOM RT 2009 speaker diarization system: enhancements in speaker modelling and cluster purification. In: Proceedings ICASSP, pp. 4958–4961. IEEE (2010)
2. Canavan, A., Graff, D., Zipperlen, G.: CALLHOME American English speech, LDC97S42. In: LDC Catalog, Linguistic Data Consortium, Philadelphia (1997)
3. Church, K., Zhu, W., Vopicka, J., Pelecanos, J., Dimitriadis, D., Fousek, P.: Speaker diarization: a perspective on challenges and opportunities from theory to practice. In: Proceedings ICASSP, pp. 4950–4954 (2017)
4. Dehak, N., Kenny, P., Dehak, R., Dumouchel, P., Ouellet, P.: Front-end factor analysis for speaker verification. IEEE Trans. Audio Speech Lang. Process. **19**(4), 788–798 (2011)
5. Fergani, B., Davy, M., Houacine, A.: Speaker diarization using one-class support vector machines. Speech Commun. **50**(5), 355–365 (2008)
6. Garcia-Romero, D., Snyder, D., Sell, G., Povey, D., McCree, A.: Speaker diarization using deep neural network embedings. In: Proceedings ICASSP, pp. 4930–4934 (2017)
7. Gupta, V.: Speaker change point detection using deep neural nets. In: Proceedings ICASSP, pp. 4420–4424 (2015)
8. Hrúz, M., Zajíc, Z.: Convolutional neural network for speaker change detection in telephone speaker diarization system. In: Proceedings ICASSP, pp. 4945–4949 (2017)
9. Lapidot, I., Bonastre, J.F.: On the importance of efficient transition modeling for speaker diarization. In: Proceedings Interspeech, 08–12 September 2016, pp. 2190–2193 (2016)
10. NIST: The 2009 (RT-09) rich transcription meeting recognition evaluation plan (2009). http://www.itl.nist.gov/iad/mig/tests/rt/2009/docs/rt09-meeting-eval-plan-v2.pdf
11. Rouvier, M., Dupuy, G., Gay, P., Khoury, E., Merlin, T., Meignier, S.: An open-source state-of-the-art toolbox for broadcast news diarization. In: Proceedings Interspeech, pp. 1477–1481 (2013)
12. Sell, G., Garcia-Romero, D.: Speaker diarization with PLDA i-vector scoring and unsupervised calibration. In: IEEE Spoken Language Technology Workshop, pp. 413–417 (2014)

13. Senoussaoui, M., Kenny, P., Stafylakis, T., Dumouchel, P.: A study of the cosine distance-based mean shift for telephone speech diarization. IEEE/ACM Trans. Audio Speech Lang. Process. **22**(1), 217–227 (2014)
14. Shum, S., Dehak, N., Chuangsuwanich, E., Reynolds, D., Glass, J.: Exploiting intra-conversation variability for speaker diarization. In: Proceedings Interspeech, pp. 945–948 (2011)
15. Wang, R., Gu, M., Li, L., Xu, M., Zheng, T.F.: Speaker segmentation using deep speaker vectors for fast speaker change scenarios. In: Proceedings ICASSP, pp. 5420–5424 (2017)
16. Zajíc, Z., Kunešová, M., Radová, V.: Investigation of segmentation in i-vector based speaker diarization of telephone speech. In: Ronzhin, A., Potapova, R., Németh, G. (eds.) SPECOM 2016. LNCS (LNAI), vol. 9811, pp. 411–418. Springer, Cham (2016). doi:10.1007/978-3-319-43958-7_49
17. Zajíc, Z., Machlica, L., Müller, L.: Initialization of fMLLR with sufficient statistics from similar speakers. In: Habernal, I., Matoušek, V. (eds.) TSD 2011. LNCS (LNAI), vol. 6836, pp. 187–194. Springer, Heidelberg (2011). doi:10.1007/978-3-642-23538-2_24
18. Zajíc, Z., Hrúz, M., Müller, L.: Speaker diarization using convolutional neural network for statistics accumulation refinement. In: Proceedings Interspeech (2017, in press)
19. Zhu, W., Pelecanos, J.: Online speaker diarization using adapted i-vector transforms. In: Proceedings ICASSP, pp. 5045–5049. IEEE (2016)