MULTI-CLASS SPECTRAL CLUSTERING WITH OVERLAPS FOR SPEAKER DIARIZATION

Desh Raj[1], *Zili Huang*[1], *Sanjeev Khudanpur*[1,2]

[1]Center for Language and Speech Processing & [2]Human Language Technology Center of Excellence
The Johns Hopkins University, Baltimore, MD 21218, USA.
draj@cs.jhu.edu, {hzili1,khudanpur}@jhu.edu

ABSTRACT

This paper describes a method for overlap-aware speaker diarization. Given an overlap detector and a speaker embedding extractor, our method performs spectral clustering of segments under overlap constraints. This is achieved by transforming the discrete clustering problem into a relaxed optimization problem which is solved by eigen-decomposition. Thereafter, we discretize the solution by alternatively using singular value decomposition and a modified version of non-maximal suppression which is constrained by the output of the overlap detector. Furthermore, we detail an HMM-DNN based overlap detector which performs frame-level classification and enforces constraints through HMM state transitions. Our method achieves a test DER of 24.0% on the mixed-headset setting of the AMI meeting corpus, which is a relative improvement of 15.2% over a strong agglomerative hierarchical clustering baseline, and compares favorably with other overlap-aware diarization methods. Further analysis on the LibriCSS data demonstrates the effectiveness of the proposed method in high overlap conditions.

Index Terms— speaker diarization, overlap detection, spectral clustering, optimization

1. INTRODUCTION

Speaker diarization (or "who spoke when?") refers to the task of segmenting speech into homogeneous speaker-specific regions [1, 2]. Conventional diarization systems [3, 4] consist of four major components. First, a speech activity detection module removes the non-speech segments. Next, the speech regions of the recording are divided into small (often overlapping) segments and a pretrained speaker embedding extractor is used to obtain fixed-dimensional embeddings, such as i-vectors, or neural embeddings [5, 6, 7], for each segment. The embeddings are scored pairwise using a cosine or probabilistic linear discriminant analysis (PLDA) similarity metric, and clustering (agglomerative or spectral) is performed on the resulting affinity matrix until a stopping criterion is reached or until the desired number of speaker clusters are obtained.

Finally, a resegmentation module [8] may be used for frame-level refinement of the clustering output.

Although this approach has proved to be effective through use of deep neural network based speaker embeddings, it does not handle overlapping speaker segments, since the clustering process assigns each segment to exactly one speaker. Existing approaches to solve the overlap problem fall into two categories. In the first framework, an externally trained overlap detection module identifies frames in the recording which contain overlapping speech. This "overlap detection" may be performed using hidden Markov models (HMMs) [9, 10, 11] or neural networks [12, 13, 14, 15]. Once overlaps are detected, an "overlap assignment" stage assigns additional speaker labels to the overlapping frames. Recently, [16] proposed overlap-aware resegmentation, which leverages the variational Bayes (VB)-HMM method used originally for diarization in [17], and applied to resegmentation in [8]. In the second framework, end-to-end systems [18, 19] are used to perform overlapping diarization in a supervised setting.

In this paper, we focus on the former approach for overlap-aware speaker diarization. Specifically, we train an external overlap detector, and use its classification decision during clustering of the segment-level embeddings. Our method relies on the two-step clustering formulation proposed in [20]. In the first step, the NP-hard discrete clustering problem is relaxed into a continuous version by ignoring the discrete constraints on the solution. The continuous problem thus obtained has a solution set generated through orthonormal transformations of eigenvectors of the normalized Laplacian. The second step involves "optimal discretization", which finds a discrete solution under the constraint that it is close (in Frobenius norm) to any of the relaxed solutions from the solution set obtained previously. We introduce overlap awareness in the discretization stage by modifying the "sum-to-one" constraint in this subproblem. This modification makes it possible to perform overlap-aware spectral clustering at no extra computational cost beyond computing the overlap decisions. Furthermore, we use the recently proposed p-binarization and normalized maximum eigengap (NME) techniques [21] to self-tune the clustering process,

thus requiring no hyperparameter tuning for estimating the number of speakers.

The remainder of this paper is organized as follows. We start by giving a detailed description of our method in Section 2, where we discuss p-binarization and NME for estimating the number of speakers followed by the mathematical formulation of multi-class spectral clustering. We then introduce our modification of the method to perform overlap-aware diarization. We describe our HMM-DNN based overlap detector in Section 3. This is followed by a description of our experimental setup and results in Sections 4 and 5, respectively. We present results on the AMI meeting corpus and the LibriCSS dataset, with detailed analysis on the performance of the method on different overlap conditions. In Section 6, we summarize previous work on spectral clustering for speaker diarization. We conclude with a discussion of future work in Section 7. In the interest of reproducible research, we discuss our implementation in detail in Section 4.3, and our code will be made publicly available[1].

2. METHODOLOGY

Our diarization follows the conventional clustering method studied extensively in previous work. We first obtain speech regions from the recording using an oracle speech activity detector (although any SAD can be used for this purpose). These are divided into small (overlapping) segments using a sliding window method, and embeddings are extracted for each segment using an x-vector extractor. Subsequently, these embeddings are clustered to obtain speaker groupings, and each cluster is labeled as a speaker.

We do not present the details of the x-vector extraction here; readers unfamiliar with how they are trained are referred to [7]. Our focus is on the final clustering stage, and particularly on how to make the clustering process overlap-aware. In particular, given a sequence of segment embeddings $U = (\mathbf{u}_1, \ldots, \mathbf{u}_i, \ldots, \mathbf{u}_N)$, the objective is to compute a label sequence $L = (\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_i, \ldots, \boldsymbol{\ell}_N)$, where $\boldsymbol{\ell}_i$ may be a single label or, in case of overlapping segment, a tuple of multiple labels. For convenience, we assume that overlaps can occur between at most two speakers, so $\boldsymbol{\ell}_i$ will be a 2-tuple for an overlapping segment. Finally, suppose we have an overlap detector which decides, for each segment, whether or not it contains overlapping speech. We denote this as

$$f(U) = \mathbf{v}_{OL}, \tag{1}$$

where $\mathbf{v}_{OL} \in \{0, 1\}^N$, and $\mathbf{v}_{OL}^i = 1$ indicates that $\boldsymbol{\ell}_i$ is an n-tuple with $n > 1$. Given U and $\mathbf{v}_{OL}$, overlap-aware diarization seeks to compute an optimal label sequence L which minimizes the diarization error. Since we do not additionally have information about the number of speakers K in the recording, we first estimate it using the heuristic described

in [21]. Subsequently, we perform multi-class spectral clustering to group the N segments into the estimated $\widehat{K}$ clusters using the optimal discretization procedure proposed in [20], where we make a key modification to constrain the optimization process on the output $\mathbf{v}_{OL}$ of our overlap detector.

2.1. Estimating number of speakers

We first use the method described in [21] to estimate the number of speakers $\widehat{K}$ in the recording using an eigengap heuristic. Here, we describe this method briefly.

Given U, we compute the affinity matrix $\mathbf{A} \in [-1, 1]^{N \times N}$ of raw cosine similarity values. Then, p-binarization is performed on this matrix by replacing the p highest similarity values in each row with 1, and the rest with 0, followed by a symmetrization operation,

$$\bar{\mathbf{A}}_p = \frac{1}{2}(\mathbf{A}_p + \mathbf{A}_p^T). \tag{2}$$

We compute the unnormalized Laplacian for this matrix,

$$\mathbf{L}_p = \mathbf{D}_p - \bar{\mathbf{A}}_p, \tag{3}$$

where $\mathbf{D}_p = \mathrm{diag}\{d_1, \ldots, d_N\}$, $d_i = \sum_{n=1}^{N} a_{in}$, also known as the "degree" of node i.

The properties of the unnormalized Laplacian of the affinity matrix have been studied extensively [22], and it is known that $\mathbf{L}_p$ has N non-negative, real eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_N$. Furthermore, an implication of the Davis-Kahan perturbation theory [23] proposes an eigengap heuristic for the optimal number of clusters. In [21], the authors used this heuristic to estimate the optimal p value for the binarization (described earlier). Specifically, let $\mathbf{e}_p$ denote the vector of differences in consecutive eigenvalues ordered in increasing order. We compute the quantities

$$g_p = \frac{\max(\mathbf{e}_p)}{\lambda_{p,N} + \epsilon}, \quad \text{and} \quad r(p) = \frac{p}{g_p}. \tag{4}$$

Then, the optimal p, i.e. $\hat{p}$ is the one that minimizes $r(p)$, and subsequently, the optimal number of clusters is given as

$$\widehat{K} = \arg\max(\mathbf{e}_{\hat{p}}). \tag{5}$$

In the multi-class spectral clustering procedure below, we will use this estimate $\widehat{K}$ for the number of clusters, and drop the subscript p from the matrices like $\mathbf{L}_p$, $\mathbf{D}_p$ and $\bar{\mathbf{A}}_p$ for brevity.

2.2. Multi-class spectral clustering

Bipartite graph partitioning using the affinity matrix Laplacian $\mathbf{L}$ is solved by node assignment based on the underlying Fiedler vector (eigenvector corresponding to the second smallest eigenvalue) [24]. The Ng-Jordan-Weiss algorithm [25] is a popular extension of this principle for multi-way partitioning of the graph. It applies K-means clustering

on the first K eigenvectors of L, i.e., in the K-eigenspace of the Laplacian. It is known that if the original samples are separable into K groups using some transformation, then their projection on the K-eigenspace can be easily grouped using K-means clustering. Recent work on speaker diarization through spectral clustering of x-vectors, as in [21] and [26], has employed this algorithm. However, there are two major limitations of this approach. First, the K-means clustering process may get stuck in bad local optima, particularly when the affinity matrix is noisy. Second, and particularly relevant for our case, it is difficult to extend this method to handle overlaps. To remedy these issues, we use an alternative formulation of spectral clustering, proposed in [20].

Given $\mathbf{A}$ and $\mathbf{D}$ as defined earlier, the clustering problem requires estimating the assignment matrix X. In graph partitioning terms, this can be represented as

$$\begin{aligned}
\text{maximize} \quad & \epsilon(X) = \frac{1}{K} \sum_{k=1}^{K} \frac{X_k^T \mathbf{A} X_k}{X_k^T \mathbf{D} X_k} \\
\text{subject to} \quad & X \in \{0,1\}^{N \times K}, \\
& X \mathbf{1}_K = \mathbf{1}_N.
\end{aligned} \quad (6)$$

Intuitively, the objective function $\epsilon(X)$ seeks to maximize the average "link-ratio", i.e., the fraction of all link weights in a group that stay within the group. The constraint $X\mathbf{1}_K = \mathbf{1}_N$ enforces the condition that each sample can belong to exactly 1 cluster. We will see later (cf. Section 2.3) how this constraint can be modified for our overlap-aware scenario.

The optimization problem in (6) is NP-complete due to the discrete constraints on X. Instead of solving this original problem, we solve a relaxed version of this problem which ignores the constraints. Let

$$Z = f(X) = X(X^T \mathbf{D} X)^{-\frac{1}{2}}. \quad (7)$$

It is easy to verify that $Z^T \mathbf{D} Z = I_K$. We can rewrite the above problem (6), by ignoring the constraints, as

$$\begin{aligned}
\text{maximize} \quad & \epsilon(Z) = \text{tr}(Z^T \mathbf{A} Z) \\
\text{subject to} \quad & Z^T \mathbf{D} Z = I_K.
\end{aligned} \quad (8)$$

Since Z has been relaxed into the continuous domain, the new optimization problem becomes tractable. Let

$$P = \mathbf{D}^{-1} \mathbf{A}, \quad (9)$$

and suppose the eigen-decomposition of P is given as $PV = VS$. Let $\Lambda^* = \text{diag}(s_1, \ldots, s_K)$ and Z^* contain the first K columns of V. Then the global optimum of the problem described in equation (8) occurs at

$$\{Z^* R : R^T R = I_K, P Z^* = Z^* \Lambda^*\}. \quad (10)$$

This implies that the global optimum is not unique; rather, it is a subspace spanned by the first K eigenvectors of P through orthonormal matrices.

The matrix Z is a continuous solution to our clustering problem. To obtain a discrete solution, we solve for a discrete approximation for Z. First, we note from equation (7) that

$$X = f^{-1}(Z) = \text{Diag}\left(\text{diag}^{-\frac{1}{2}}\left(ZZ^T\right)\right) Z. \quad (11)$$

Using this transformation, we can characterize the solution obtained in equation (10) as

$$\{\tilde{X}^* R : R^T R = I_K, \tilde{X}^* = f^{-1}(Z^*)\}. \quad (12)$$

Now, our discretization problem is to find an X which approximates $\tilde{X}^* R$ for some orthonormal R, such that X obeys the discrete constraints from problem (6). Mathematically, this is formulated as

$$\begin{aligned}
\text{minimize} \quad & \phi(X, R) = \left\| X - \tilde{X}^* R \right\|^2 \\
\text{subject to} \quad & X \in \{0,1\}^{N \times K}, \\
& X \mathbf{1}_K = \mathbf{1}_N, \\
& R^T R = I_K.
\end{aligned} \quad (13)$$

It is difficult to minimize $\phi(X, R)$ jointly in X and R, so we optimize it alternately in X and R. Suppose we are given some R^*, then the problem (13) reduces to

$$\begin{aligned}
\text{minimize} \quad & \phi(X) = \left\| X - \tilde{X}^* R^* \right\|^2 \\
\text{subject to} \quad & X \in \{0,1\}^{N \times K}, \\
& X \mathbf{1}_K = \mathbf{1}_N.
\end{aligned} \quad (14)$$

The solution to this problem is given by non-maximal suppression, i.e.,

$$X^*(i, l) = \left\langle l = \arg\max_{k \in [K]} \tilde{X}(i, k) \right\rangle, \quad i \in \{1, \ldots, N\}. \quad (15)$$

Intuitively, we set the largest entry in each row as 1 and zero out all the others. This ensures that each sample belongs to exactly 1 cluster. In the next section, we will see how to reformulate problem (14) for the case when some samples can belong to more than one clusters.

Next, we fix X^* and solve the following problem for R^*:

$$\begin{aligned}
\text{minimize} \quad & \phi(R) = \left\| X^* - \tilde{X}^* R \right\|^2 \\
\text{subject to} \quad & R^T R = I_K.
\end{aligned} \quad (16)$$

The solution to this problem is given by

$$R^* = \tilde{U} U^T, \quad (17)$$

where $(U, \Omega, \tilde{U})$ is a singular value decomposition of $X^{*T} \tilde{X}^*$.

We solve the two problems (14) and (16) iteratively until convergence, and finally return X^* as the output of the clustering procedure.

2.3. Extension to overlap-aware clustering

From equation (1), suppose the output of our overlap detector is given by $\mathbf{v}_{OL}$. Then, we can reformulate problem (14) to include the overlap constraint as

$$
\begin{aligned}
\text{minimize} \quad & \phi(X) = \left\| X - \tilde{X}^* R^* \right\|^2 \\
\text{subject to} \quad & X \in \{0,1\}^{N \times K}, \\
& X\mathbf{1}_K = \mathbf{1}_N + \mathbf{v}_{OL}.
\end{aligned}
\tag{18}
$$

Intuitively, this solves the same optimum discretization problem, but the cluster exclusivity constraint has been modified to represent the condition that some samples may belong to more than one cluster. Similar to how we used non-maximal suppression in (15) to solve the problem previously, the solution to the modified problem is again given by non-maximal suppression, with the exception that for samples belonging to more than one group, we set the largest two entries to 1, while zeroing out the others. Mathematically,

$$
\hat{X}^*(i,l) = X^*(i,l) + \mathbf{v}_{OL}^{(i)} \times \langle l = k_i' \rangle,
\tag{19}
$$

$\forall i \in \{1, \ldots, N\}$, where $X^*(i,l)$ is defined in (15), and k_i' is the index of the second largest element of $\tilde{X}_i$. Although we only consider the case of 2-speaker overlaps here, it is easy to extend this method to the case of an arbitrary number of overlapping speakers.

3. OVERLAP DETECTION

Our proposed overlap-aware diarization method relies heavily on the performance of $f(U)$, the overlap detector (OD). In this section, we detail an HMM-DNN based overlap detector. Our model is similar to the speech activity detector previously used in the CHiME-6 baseline system [27].

We first trained a neural network classifier to assign each frame in an utterance a label from $\mathcal{C} = \{$silence, single, overlap$\}$, denoting silence, single speaker, or overlapping regions, respectively. We used the architecture shown in Figure 1, consisting of time-delay neural network (TDNN) layers to capture long temporal contexts [28], interleaved with bidirectional long short term memory (BLSTM) layers with projection, to incorporate utterance-level statistics.

The posteriors obtained from the classifier were scaled with an external bias parameter tuned on the development data to reduce the false alarm rate. We then post-processed the per-frame classifier outputs to enforce minimum and maximum silence/single/overlap durations, by constructing a simple HMM whose state transition diagram encodes these constraints. Treating the per-frame posteriors like emission probabilities, we performed Viterbi decoding to obtain the most likely label-sequence. Furthermore, state transitions between the silence and overlap states were prohibited, mimicking real-world observations where it is highly unlikely for two speakers to start or stop speaking simultaneously.
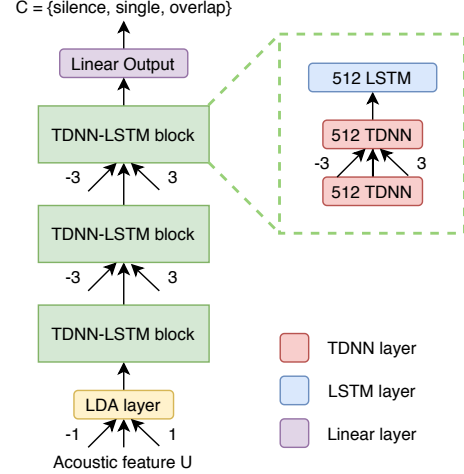

Fig. 1: Architecture of the neural network used for frame-level classification for overlap detection.

4. EXPERIMENTAL SETUP

4.1. Datasets

We performed experiments on two datasets – the AMI meeting corpus [29], and the LibriCSS data [30]. AMI consists of 100 hours of recorded meetings containing 4 speakers per session, with speech from close-talk, single distant microphone (SDM), and array microphones. For our experiments, we used the mix-headset recordings, which are obtained by summing the individual headset signals from the participants in the meeting. The dataset contains approximately 20% overlap ratio, i.e., 20% of the total speech contains overlaps. LibriCSS is a recently released corpus consisting of multichannel audio recordings of "simulated conversations." It comprises 10 sessions, where each session is approximately one hour long. Each session is made up of six 10-minute-long "mini sessions" that have different overlap ratios, ranging from 0 to 40%, and contain 8 speakers. The recordings were made in a regular meeting room by using a seven-channel circular microphone array. For our experiments, we selected the recordings from the first channel of the array. We used this dataset to conduct a performance analysis of our proposed method on different overlap conditions.

4.2. Baselines

We first have single-speaker baselines: (i) agglomerative hierarchical clustering (AHC) of x-vectors with probabilistic linear discriminant analysis (PLDA) scoring [31], (ii) spectral clustering of x-vectors with cosine scoring (using the Ng-Jordan-Weiss method) [21], and (iii) Bayesian HMM based x-vector clustering (VBx) [32, 33]. We used the same x-vector extractor for all the baselines (described in Section 4.3), such that the difference in their performance was only due to the clustering process. Furthermore, we used the same PLDA

model (trained on a subset of the AMI training data) for the AHC and VBx baselines[2]. For both these baselines, hyperparameters were tuned on the development set. No hyperparameter selection is required for spectral clustering since it is auto-tuned. We used a ground-truth VAD for these baselines as well as for our proposed method.

We also compare our approach with diarization methods that are not overlap-agnostic. These include: (i) overlap-aware VB resegmentation [16] and (ii) region proposal networks (RPNs) [19]. For the former, we present the official results from the paper, which uses a neural VAD. For RPNs, we filtered out non-speech regions using the ground truth VAD.

4.3. Implementation details

X-vector extractor. We used a similar x-vector extractor as described in earlier studies [3, 21]. The model consists of TDNN layers with statistics pooling, and we extracted 128-dim embeddings from the pre-final layer. It was trained on VoxCeleb data [34] with simulated room impulse responses [35] using the Kaldi toolkit [36], and released as part of the CHiME-6 baseline [27].

Overlap detector. We trained an HMM-DNN overlap detector (described in Section 3) using Kaldi. We used 40-dim MFCCs features as input, and trained the classifier on in-domain training data. For AMI, we used targets obtained from annotations of the official training set. Since LibriCSS does not have a corresponding training data, we generated simulated mixtures with reverberation using the Librispeech training utterances and used force-aligned targets for training our overlap detector. The decoding graph was created with additional constraints on the minimum (maximum) durations for single speaker and overlapping regions as 0.03s (10.0s) and 0.1s (5.0s), respectively. Note that our clustering method itself is independent of the overlap detector used.

Overlap-aware spectral clustering. We extended the spectral clustering algorithm in scikit-learn [37] for our implementation. Since the overlap detector provides frame-level classification decisions whereas x-vectors were extracted for 1.5s segments, we assumed that a segment is "overlapping" if at least half of it lies in overlapping regions. For estimating the number of speakers $\widehat{K}$, we swept the binarization factor p in the range from 2 to 20, similar to what was done in [21].

5. RESULTS AND DISCUSSION

5.1. Overlap detection on AMI

We present the results obtained by our overlap detector using 40-dim MFCC features on AMI mix-headset data in Table 1. We can see that the performance is comparable to previous

---

Table 1: Overlap detection results on AMI mix-headset data, in terms of Precision (%) and Recall (%).

Model (feature type)	Dev		Eval	
	Precision	Recall	Precision	Recall
ConvNet (Spectogram) [15]	80.5	50.2	75.8	44.6
E2E BLSTM (MFCC) [16]	90.0	52.5	91.9	48.4
E2E BLSTM (SincNet) [16]	90.0	63.8	86.8	65.8
Our method (MFCC)	83.9	68.5	86.4	65.2

Table 2: Diarization results on AMI mix-headset eval set, in terms of missed speech (MS), false alarm (FA), speaker confusion (Conf.), and diarization error rate (DER).

Method	MS	FA	Conf.	DER
AHC/PLDA	19.9	0.0	8.4	28.3
Spectral/cosine	19.9	0.0	7.0	26.9
VBx [32]	19.9	0.0	6.3	26.2
VB resegmentation [16]	13.0	3.6	7.2	23.8
RPN [19]	9.5	7.7	8.3	25.5
Our method	11.3	2.2	10.5	24.0
Our method + oracle OD	7.4	1.8	12.3	<u>21.5</u>
Our method + noise aug.	11.3	2.2	10.1	**23.6**

studies on this dataset, without using waveform-level learned features. We used the output from this overlap detector for further experiments.

5.2. Diarization results for AMI

Table 2 shows our proposed overlap-aware spectral clustering method compared with baselines, evaluated on the AMI mix-headset eval data. Using our overlap detector trained on the AMI train set, we were able to improve the DER from 28.3% for the AHC/PLDA baseline, to 24.0%, which is a relative improvement of 15.2%. This compares favorably with the performance of other diarization methods like overlap-aware VB resegmentation and RPNs. Furthermore, it is possible to reduce the DER to 21.5% using an oracle overlap detector.

A detailed analysis of the results reveals that although the missed speech reduces substantially (from 19.9% to 11.3%) as a result of overlap detection, there is also a significant increase in speaker confusion errors (from 8.4% to 10.5%). We conjecture that since the x-vector extractor was trained only on single-speaker utterances, a mismatch in the overlap regions of the recording results in noisy samples. The speaker confusion improved by 0.4% when we used an x-vector extractor trained with noise augmentation, using noises from the MUSAN corpus [38]. To verify our hypothesis further, we show the T-SNE plots for the non-overlapping and overlapping segments in Fig. 2(a) and 2(b), respectively. We can see that while the embeddings for the non-overlapping segments are well separated, those for overlapping segments may often
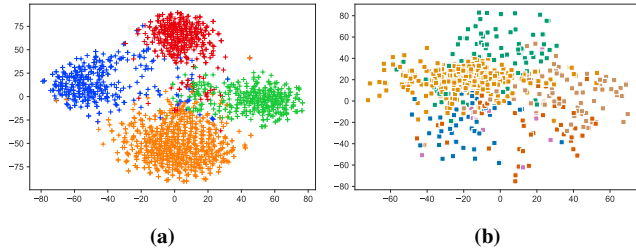
Fig. 2: X-vector embeddings for (a) non-overlapping, and (b) over-lapping segments for the recording `EN2002a` in the AMI eval set (containing 4 speakers). Colors denote the speaker assigned to the segment. For (b), each color represents a distinct pair of speaker labels, resulting in 6 differently colored clusters.

Table 3: Diarization performance on LibriCSS evaluation set (sessions 2-10), evaluated condition-wise, in terms of % DER. 0S and 0L refer to 0% overlap with short and long inter-utterance silences, respectively. Our overlap detector obtained 96.3% precision and 83.8% recall on this data. RPN does not use ground truth VAD.

Method	Overlap ratio in %						Average
	0L	0S	10	20	30	40	
AHC/PLDA	5.1	2.7	11.5	19.3	22.7	28.8	16.3
Spectral/cosine	2.2	1.9	8.5	13.5	19.0	23.4	12.6
RPN	4.5	9.1	8.3	6.7	11.6	14.2	9.5
Our method	2.6	3.4	6.8	10.0	13.9	15.2	9.3
+ oracle OD	2.2	3.3	6.7	9.6	12.9	14.4	8.8

be noisy, leading to clustering errors.

5.3. Analysis on LibriCSS

Table 3 shows a breakdown of diarization errors obtained by our system, compared with some of the baselines. It is evident that as the overlap ratio increases from 0 to 40%, the difference in performance becomes more significant. On average, our method provided a 42.9% relative DER improvement compared to a baseline AHC system, and this increased to 46.0% relative on using an oracle overlap detector. We note here that since LibriCSS does not have a corresponding training set, the PLDA was trained on Librispeech utterances. The mismatch between clean training data versus overlapping mixtures at test time may be particularly detrimental to the performance of the AHC system.

6. RELATED WORK

Spectral clustering was first applied to speaker diarization in [39] using the Ng-Jordan-Weiss (NJW) algorithm [25]. [40] extended this to the case of an unknown number of speakers by using the eigengap criterion. Agglomerative and spectral clustering methods for meeting diarization were compared in [41]. After i-vectors were proposed for speaker recognition [5], they were combined with cosine scoring

and spectral clustering to perform diarization in [42]. It was further observed that spectral clustering was more robust to non-stationary environmental noise compared to other clustering methods [43]. More recently, with the ubiquitousness of deep neural networks, several researchers have proposed methods to incorporate DNNs with spectral clustering. [44] proposed a supervised method to measure the similarity matrix between all segments of an audio recording with BLSTMs, and applied spectral clustering on top of the similarity matrix. Other approaches use DNN-based speaker embeddings, such as x-vectors [7], to compute the similarty matrix between segment pairs [21, 26]. Additionally, [21] introduced p-binarization and normalized maximum eigengap (NME) techniques to automatically estimate the number of speakers in the recording.

Speaker diarization in overlapping settings has been studied extensively (cf. Section 1). However, to the best of our knowledge, there is no prior work on incorporating overlap awareness into spectral clustering based diarization. The recently proposed target speaker voice activity detection (TS-VAD) [26] method uses x-vector based spectral clustering for initial estimate of speaker i-vectors, and thereafter performs frame-level multi-label classification to predict speaker activities in a speech frame.

7. CONCLUSION

We proposed a new method for overlap-aware speaker diarization using spectral clustering. We leveraged an external overlap detector to identify the overlapping subsegments, and then assigned these segments to multiple speakers during clustering. The clustering approach itself was reformulated by first relaxing the discrete constraints, and then solving an optimal discretization problem with the additional overlap constraints. Our method provided significant improvements over conventional single-speaker clustering models on the AMI meeting corpus, and was competitive with other overlap-aware diarization methods. Analysis on LibriCSS showed that overlapping regions benefit strongly from this approach, although speaker confusion may increase due to an inadequate speaker embedding extractor. We conjecture that this problem may be alleviated by training the x-vector extractor additionally on overlapping segments. This investigation is left as future work.

8. ACKNOWLEDGMENT

9. REFERENCES

[1] Xavier Anguera Miró, Simon Bozonnet, Nicholas W. D. Evans, Corinne Fredouille, Gerald Friedland, and Oriol Vinyals, "Speaker diarization: A review of recent research," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, pp. 356–370, 2012.

[2] Sue Tranter and Douglas A. Reynolds, "An overview of automatic speaker diarization systems," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, pp. 1557–1565, 2006.

[3] Daniel Garcia-Romero, David Snyder, Gregory Sell, Daniel Povey, and Alan McCree, "Speaker diarization using deep neural network embeddings," *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4930–4934, 2017.

[4] Lei Sun, Jun Du, Chao Jiang, Xueyang Zhang, Shan He, Bing Yin, and Chin-Hui Lee, "Speaker diarization with enhancing speech for the first dihard challenge," in *INTERSPEECH*, 2018.

[5] Najim Dehak, Patrick Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, pp. 788–798, 2011.

[6] Ehsan Variani, Xin Lei, Erik McDermott, Ignacio Lopez-Moreno, and Javier Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," *ICASSP*, pp. 4052–4056, 2014.

[7] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5329–5333, 2018.

[8] Gregory Sell and Daniel Garcia-Romero, "Diarization resegmentation in the factor analysis subspace," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4794–4798, 2015.

[9] Kofi Boakye, Beatriz Trueba-Hornero, Oriol Vinyals, and Gerald Friedland, "Overlapped speech detection for improved speaker diarization in multiparty meetings," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2008, pp. 4353–4356.

[10] Marijn Huijbregts, David A. van Leeuwen, and Franciska de Jong, "Speech overlap detection in a two-pass speaker diarization system," in *INTERSPEECH*, 2009.

[11] Sree Harsha Yella and Fabio Valente, "Speaker diarization of overlapping speech based on silence distribution in meeting recordings," in *INTERSPEECH*, 2012.

[12] Jürgen T. Geiger, Florian Eyben, Björn W. Schuller, and Gerhard Rigoll, "Detecting overlapping speech with long short-term memory recurrent neural networks," in *INTERSPEECH*, 2013.

[13] Valentin Andrei, Horia Cucu, and Corneliu Burileanu, "Detecting overlapped speech on short timeframes using deep learning," in *INTERSPEECH*, 2017.

[14] Gerhard Hagerer, Vedhas Pandit, Florian Eyben, and Björn W. Schuller, "Enhancing LSTM RNN-based speech overlap detection by artificially mixed data," in *Semantic Audio*, 2017.

[15] Marie Kunesová, Marek Hrúz, Zbynek Zajíc, and Vlasta Radová, "Detection of overlapping speech for the purposes of speaker diarization," in *SPECOM*, 2019.

[16] Latané Bullock, Hervé Bredin, and L. Paola García-Perera, "Overlap-aware diarization: resegmentation using neural end-to-end overlapped speech detection," *ArXiv*, vol. abs/1910.11646, 2019.

[17] Mireia Díez, Lukás Burget, and Pavel Matejka, "Speaker diarization based on Bayesian HMM with eigenvoice priors," in *Odyssey*, 2018.

[18] Yusuke Fujita, Shinji Watanabe, Shota Horiguchi, Yawen Xue, and Kenji Nagamatsu, "End-to-end neural diarization: Reformulating speaker diarization as simple multi-label classification," *ArXiv*, vol. abs/2003.02966, 2020.

[19] Zili Huang, Shinji Watanabe, Yusuke Fujita, Paola García, Yiwen Shao, Daniel Povey, and Sanjeev Khudanpur, "Speaker diarization with region proposal network," *ArXiv*, vol. abs/2002.06220, 2020.

[20] Stella X. Yu and Jianbo Shi, "Multiclass spectral clustering," *Proceedings Ninth IEEE International Conference on Computer Vision*, pp. 313–319 vol.1, 2003.

[21] Tae Jin Park, Kyu J. Han, Manoj Kumar, and Shrikanth S. Narayanan, "Auto-tuning spectral clustering for speaker diarization using normalized maximum eigengap," *IEEE Signal Processing Letters*, vol. 27, pp. 381–385, 2020.

[22] Ulrike von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, pp. 395–416, 2007.

[23] G. W. Stewart and Ji-Guang Sun, "Matrix perturbation theory," 1990.

[24] Miroslav Fiedler, "Algebraic connectivity of graphs," 1973.

[25] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss, "On spectral clustering: Analysis and an algorithm," in *NIPS*, 2001.

[26] Ivan Medennikov, Maxim Korenevsky, Tatiana Prisyach, Yuri Y. Khokhlov, Mariya Korenevskaya, Ivan Sorokin, Tatiana V. Timofeeva, Anton Mitrofanov, Andrei Andrusenko, Ivan Podluzhny, Aleksandr Laptev, and Aleksei Romanenko, "Target-speaker voice activity detection: a novel approach for multi-speaker diarization in a dinner party scenario," *ArXiv*, vol. abs/2005.07272, 2020.

[27] Shinji Watanabe, Michael Mandel, Jon Barker, and Emmanuel Vincent, "Chime-6 challenge: Tackling multispeaker speech recognition for unsegmented recordings," *ArXiv*, vol. abs/2004.09249, 2020.

[28] Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *INTERSPEECH*, 2015.

[29] Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Maël Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, Guillaume Lathoud, Mike Lincoln, Agnes Lisowska Masson, Iain McCowan, Wilfried Post, Dennis Reidsma, and Pierre Wellner, "The ami meeting corpus: A pre-announcement," in *MLMI*, 2005.

[30] Zhuo Chen, Takuya Yoshioka, Liang Lu, Tianyan Zhou, Zhong Meng, Yi Luo, J. Wu, and Jinyu Li, "Continuous speech separation: Dataset and analysis," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7284–7288, 2020.

[31] Gregory Sell, David Snyder, Alan McCree, Daniel Garcia-Romero, Jesús Villalba, Matthew Maciejewski, Vimal Manohar, Najim Dehak, Daniel Povey, Shinji Watanabe, and Sanjeev Khudanpur, "Diarization is hard: Some experiences and lessons learned for the jhu team in the inaugural dihard challenge," in *INTERSPEECH*, 2018.

[32] Mireia Diez, Lukás Burget, Shuai Wang, Johan Rohdin, and Jan ernocký, "Bayesian hmm based x-vector clustering for speaker diarization," in *INTERSPEECH*, 2019.

[33] Mireia Díez, Lukás Burget, Federico Landini, Shuai Wang, and Jan ernocký, "Optimizing bayesian hmm based x-vector clustering for the second dihard speech diarization challenge," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6519–6523, 2020.

[34] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman, "VoxCeleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[35] Tom Ko, Vijayaditya Peddinti, Daniel Povey, Michael L Seltzer, and Sanjeev Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *ICASSP*. IEEE, 2017, pp. 5220–5224.

[36] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The Kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011, number CONF.

[37] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[38] David Snyder, Guoguo Chen, and Daniel Povey, "Musan: A music, speech, and noise corpus," *ArXiv*, vol. abs/1510.08484, 2015.

[39] Huazhong Ning, Ming Liu, Hao Tang, and Thomas S. Huang, "A spectral clustering approach to speaker diarization," in *INTERSPEECH*, 2006.

[40] Nikoletta Bassiou, Vassiliki Moschou, and Constantine Kotropoulos, "Speaker diarization exploiting the eigengap criterion and cluster ensembles," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, pp. 2134–2144, 2010.

[41] Jordi Luque and Javier Hernando, "On the use of agglomerative and spectral clustering in speaker diarization of meetings," in *Odyssey*, 2012.

[42] Stephen Shum, Najim Dehak, and Jim Glass, "On the use of spectral and iterative methods for speaker diarization," in *INTERSPEECH*, 2012.

[43] Naohiro Tawara, Tetsuji Ogawa, and Tetsunori Kobayashi, "A comparative study of spectral clustering for i-vector-based speaker clustering under noisy conditions," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2041–2045, 2015.

[44] Qingjian Lin, Ruiqing Yin, Ming Li, Hervé Bredin, and Claude Barras, "Lstm based similarity measurement with spectral clustering for speaker diarization," in *INTERSPEECH*, 2019.