mathematics

MDPI

Article

Application of Fusion of Various Spontaneous Speech Analytics Methods for Improving Far-Field Neural-Based Diarization

Sergei Astapov [1,*], Aleksei Gusev [1,2], Marina Volkova [1,2], Aleksei Logunov [1,2], Valeriia Zaluskaia [1,2], Vlada Kapranova [1,2], Elena Timofeeva [1], Elena Evseeva [1], Vladimir Kabarov [1] and Yuri Matveev [1,2]

[1] Information Technologies and Programming Faculty, ITMO University, 197101 Saint Petersburg, Russia; gusev-a@speechpro.com (A.G.); volkova@speechpro.com (M.V.); logunov@speechpro.com (A.L.); zaluskaya@speechpro.com (V.Z.); kapranova@speechpro.com (V.K.); eptimofeeva@itmo.ru (E.T.); evseeva@itmo.ru (E.E.); vikabarov@itmo.ru (V.K.); matveev@mail.ifmo.ru (Y.M.)

[2] STC-Innovations Ltd., 194044 Saint-Petersburg, Russia

* Correspondence: sastapov@itmo.ru

Abstract: Recently developed methods in spontaneous speech analytics require the use of speaker separation based on audio data, referred to as diarization. It is applied to widespread use cases, such as meeting transcription based on recordings from distant microphones and the extraction of the target speaker's voice profiles from noisy audio. However, speech recognition and analysis can be hindered by background and point-source noise, overlapping speech, and reverberation, which all affect diarization quality in conjunction with each other. To compensate for the impact of these factors, there are a variety of supportive speech analytics methods, such as quality assessments in terms of SNR and RT60 reverberation time metrics, overlapping speech detection, instant speaker number estimation, etc. The improvements in speaker verification methods have benefits in the area of speaker separation as well. This paper introduces several approaches aimed towards improving diarization system quality. The presented experimental results demonstrate the possibility of refining initial speaker labels from neural-based VAD data by means of fusion with labels from quality estimation models, overlapping speech detectors, and speaker number estimation models, which contain CNN and LSTM modules. Such fusing approaches allow us to significantly decrease DER values compared to standalone VAD methods. Cases of ideal VAD labeling are utilized to show the positive impact of ResNet-101 neural networks on diarization quality in comparison with basic x-vectors and ECAPA-TDNN architectures trained on 8 kHz data. Moreover, this paper highlights the advantage of spectral clustering over other clustering methods applied to diarization. The overall quality of diarization is improved at all stages of the pipeline, and the combination of various speech analytics methods makes a significant contribution to the improvement of diarization quality.

Keywords: speaker diarization; spontaneous speech processing; voice activity detection; overlapping speech detection; speaker extractor models; speaker number estimation; model fusion; quality estimation; distant speech processing; artificial neural networks

Citation: Astapov, S.; Gusev, A.; Volkova, M.; Logunov, A.; Zaluskaia, V.; Kapranova, V.; Timofeeva, E.; Evseeva, E.; Kabarov, V.; Matveev, Y. Application of Fusion of Various Spontaneous Speech Analytics Methods for Improving Far-Field Neural-Based Diarization. *Mathematics* **2021**, *9*, 2998. https://doi.org/10.3390/math9232998

Academic Editors: Grigoreta-Sofia Cojocar and Adriana-Mihaela Guran

Received: 31 July 2021
Accepted: 17 November 2021
Published: 23 November 2021

1. Introduction

The widespread availability of tools for sound acquisition, as well as the cost reduction of audio data storage systems, require new methods for automatic processing. Such tasks as the generation of meeting minutes, the processing of telephone conversations, and the automatic transcription of news or entertainment programs, not only require speech recognition [1], but also involve audio annotation by speakers, which is usually referred to as speaker diarization.

Diarization is the process of partitioning an input audio stream into homogeneous segments according to the speaker identity [2]. This means that the goal is to determine who is speaking in each audio segment. Diarization can be used as a preliminary stage in speech recognition systems, in automatic translation or meeting recording transcription.

The choice of a suitable diarization scenario depends on the specific task and corresponding data. So, processed data can be characterized based on the channel specifics (telephone, microphone or microphone array), the presence of noise and the reverberation level, etc. Preliminary information can also influence the choice of diarization methods: whether the exact number of speakers is known or whether individual speech samples of their voices are available. When it comes to meeting minutes, it is helpful to know in advance if the participants can move around the room or tend to interrupt each other. All of these factors can significantly affect the quality of diarization [3].

The diarization scenario may consist of the following steps [4]. The first step usually tends to apply the voice activity detector (VAD) in order to obtain the markup of speech and non-speech segments of an audio recording. This stage can be affected by the quality of the recording or the presence of speaker interruptions, as mentioned above. Nevertheless, the accuracy in the finding of speech boundaries can affect the overall quality of diarization. The next step is the extraction of speaker features (speaker models) for each speech segment. This is necessary for the main purpose of diarization, in order to determine exactly who is speaking in a given segment. At this stage, well-proven representations of the speaker patterns are used: i-vectors obtained via factor analysis [5], x-vectors extracted using a time delay neural network [6] or other types of DNN-embeddings [7]. Having been obtained for each segment, the speaker representations are subject to clustering procedures during the third step of diarization. Pre-selected similarity metrics are used to divide speech segments into clusters and to match the speaker label to each cluster. Thus, in the output of the diarization system, the markup "who speaks when" is obtained. In our paper we consider approaches to improve the quality of diarization at each stage, with particular attention to the detection of speech boundaries.

The accuracy of speech boundary detection can be improved through the choice of an appropriate VAD method. Recently, in addition to standard energy-based voice activity detectors, neural network-based VADs are gaining popularity, which allows one to obtain better resistance to noise conditions [8]. In this paper, we consider the use of the DNN-based VAD described in [9] for a diarization task applied to the AMI Meeting Corpus [10]. To deal with the inherent problems of multi-dialogue recordings such as speaker interruptions and the simultaneous utterances of multiple speakers, we examine options for fusing VAD with other speech analytic systems. At a certain stage we apply instantaneous speaker number estimation, which we further refer to as a speaker counter (SC) model. The model resolves a classification problem, where each class represents the number of simultaneous speakers detected. If the speaker counter is analyzed in terms of two classes, "zero speakers" and "one or more speakers", it can be considered an alternative to VAD, and fusing VAD and SC can increase accuracy in locating speech boundaries. In turn, the performance of SC in conversations with frequent interruptions can be improved by fusing SC with the model which detects overlapping speech segments [11]. We refer to this model as the overlapping speech detector (OSD). The fusion of these two models allows one to tackle the problem of simultaneous speaker detection from separate perspectives.

Information about acoustic conditions in terms of the speech-to-noise ratio (SNR) and reverberation time (RT60) for each audio segment can also be used to distinguish speech and non-speech frames. We apply an automatic quality estimation (QE) system, described in [9], to cluster estimated SNR-RT60 vectors into speech and non-speech clusters and thereby retrieve an approximate voice activity markup. Although this method is not accurate, we investigate its usefulness in fusing it with the base DNN-VAD.

The next stage of the diarization process consists of generating speaker models for each segment located during the previous stages of speech detection. The algorithms at this stage can be implemented through speaker DNN-embedding extraction, similar to the speaker verification task. The current state-of-the-art systems in speaker verification are completely guided by the deep learning paradigm. Previously, the frame-level portion of these extractors was based on TDNN (time delay neural network) blocks that contained only five convolutional layers with temporal context [6]. Such types of embeddings,

referred to as x-vectors, are often applied to diarization tasks in state-of-the-art systems [12]. The newer ECAPA-TDNN (emphasized channel attention, propagation and aggregation TDNN) [13] architecture develops the idea of TDNN and contains additional blocks with hierarchical filters for the extraction of different scale features. ECAPA-TDNN and various modifications of the well-known ResNet [7] architecture are compared in our research in terms of EER, minDCF, and DER metrics.

The speaker models obtained for each speech segment must then be clustered. The purpose of clustering is to associate segments from the same speaker to one another. The clustering procedure ideally yields one cluster per each speaker in the recording, with all the segments from a given speaker contained in a single cluster. The common approach used in diarization systems is agglomerative hierarchical clustering, which can be used in the absence of prior knowledge about the number of speakers [14–16]. As an alternative method, spectral clustering [17] can also be used, as well as methods that require the specification of the number of clusters, such as KMeans or DBSCAN [18].

In this study, we aimed to enhance separate parts of the neural-based diarization system and analyze their contributions to the final assessment. In particular, our experiments were based on the following system components: feature extraction, VAD methods, speaker extractor models, speaker verification, and diarization. VAD results coupled with SC and QE estimates were applied to speaker extractor and diarization models' results through fusion algorithms in order to achieve an increase in diarization quality. Thus, separate state-of-the-art and proposed diarization system components, as well as the pipeline in their entirety, are studied and evaluated.

Since the study focuses specifically on cases of noisy overlapping speech acquired by a far-field microphone in real-life conversation conditions, it is important to take into account the same conditions when comparing our solution with those of other studies. In this sense, our proposed neural network-based VAD is compared to the publicly-available SileroVAD [19] on the evaluation subset of the Third DIHARD Speech Diarization Challenge. Our method demonstrated an ability to deal with noisy conditions and showed a 16.9% EER versus the 26.37% EER of SileroVAD. However, it is not always possible to correctly compare the results of different studies. For example, the accuracy of our proposed speaker counter detector, obtained on a realistic AMI dataset, was 65.6%, and, although this percentage was less than those presented in similar works, unlike those works, we did not employ synthesized datasets. As described below, our results in regard to diarization of the AMI evaluation set are comparable to the state-of-the-art papers, and the proposed fusion methods can be further improved.

The paper is structured as follows. Section 2 presents the datasets applied for the training and evaluation of the presented diarization pipeline and its system components and discusses the methods of feature extraction applied in these components. Section 3 describes the separate system pipeline components, namely, the approaches to VAD, SC, OSD, and QE, speaker extraction models, and the fusion methods applied to these system components. Section 4 is devoted to the experimental evaluation of the system components and the diarization pipeline. Finally, Section 5 discusses the results achieved during the study and addresses the prospects of future developments in the considered direction of research.

2. Data Processing

In this section we describe the data used to train and test all systems included in the investigated diarization pipeline. The section begins by describing the process of extracting features from raw audio as the first and primary stage of data processing.

2.1. Feature Extraction

For audio signal processing, we extract several feature types from the raw audio signal that can be fed into machine learning models. In our paper, feature extraction methods

vary depending on the diarization system units, the main components of which are speaker embedding extractor, VAD, SC, OSD, and QE models.

All speaker embedding extractors presented in this paper expect log Mel-filterbank (LMFB) energy coefficients extracted from raw input signals using the standard Kaldi recipe [20] at the sampling rates of 8 kHz or 16 kHz:

- consisting of 64 LMFB components extracted from a raw signal with a sampling rate of 8 kHz;
- consisting of 80 LMFB components extracted from a raw signal with a sampling rate of 16 kHz.

Extracted features additionally go through either one of the two different post-processing steps, depending on the type of speaker embedding extractor used afterwards:

- local cepstral mean normalization (CMN-normalization) over a 3-s sliding window;
- global cepstral mean and variance Normalization (CMVN-normalization) over the whole utterance.

The application of the above methods for each type of speaker-embedding extractor is discussed in Section 3.5.

To obtain speech segments from the audio signal, we apply a neural network-based voice activity detector (VAD) system developed by us. The model receives mel-frequency cepstral coefficients (MFCCs) extracted from the raw signal with a sampling rate of 8 kHz.

Modified versions of the voice activity detection system integrate OSD, SC or QE models with DNN-VAD. As the features, the OSD and SC neural network models use the short-term Fourier transform (STFT) coefficients extracted from the preprocessed input audio with a sampling rate of 16 kHz. The QE models use LMFB feature coefficients with CMN-normalization. The parameters of the extracted features are presented in Table 1.

Table 1. Types of applied audio features.

Model	Feature Type	Number of Coefficients	Frame Length, ms	Overlap, ms	Sampling Rate, kHz
Speaker Embedding Extractors	LMFB	64/80	25	15	8/16
VAD	MFCC	23	30	10	8
SC	STFT	201	25	10	16
OSD	STFT	81	25	10	16
QE	LMFB	64	25	15	8

2.2. Datasets

In this paper, two different sets are used for the test protocol. The target dataset for measuring the quality of diarization is the AMI corpus. This dataset is also used to train SC and OSD models, evaluate individual parts of the system (SC, OSD, clustering) and fusion. Additionally, during the study, Voxceleb1 and NIST SRE 2019 evaluation datasets are used to measure the quality of speaker verification models. In the current work, we decided to investigate the degradation of speaker extraction systems trained on the utterances sampled at 8 kHz, compared to the systems trained on files sampled at 16 kHz. For this purpose, two different datasets were created. For all datasets, the division into train/dev/test sets suggested by their original authors was used. The application of each of these datasets is discussed in detail below.

The main dataset for assessing the quality of diarization is the **AMI corpus** [10]. The dataset consists of over 100 h of meeting recordings. In general, the total number of participants in a single meeting is four (approximately 80%), and rarely three or five. The meetings were recorded in specially equipped rooms of various configurations and locations using microphone arrays, lapel microphones and headphones. In addition, each meeting attendee was provided with graphics such as videos, slides, whiteboards,

and notebooks. All recordings were synchronized. The dataset contains both the recordings of real meetings and meetings with predefined scenarios and roles. For about 70% of the recording duration only one speaker is active, whereas for about 20% of the duration speech is absent, and only 10% corresponds to the simultaneous speech of several people.

Several recent studies on the diarization problem include experiments based on the AMI corpus to measure quality. The proposed methods can differ in speaker extractor models, clustering methods, whether they include or exclude overlapping speech in the scoring, as well as whether they use references or predicted speech/non-speech labels. It is also essential to use the same evaluation protocols for the AMI database for a fair comparison, in particular, to select data from the same set of microphones: Headset-Mix, Lapel-Mix or Distant-Mic. The work [21] closest to our solution compares the x-vector TDNN and ECAPA-TDNN architectures for speaker model extraction and ignores overlapping speech segments during scoring. In this work, the best results are obtained using the ECAPA-TDNN architecture with a spectral clustering back-end, which achieved 3.01% DER for the case of the estimated number of speakers and 2.65% DER for the case of a known number of speakers on the AMI Headset-Mix evaluation set. Another work [22] compares the well-known agglomerative hierarchical clustering (AHC) method and a proposed modification of the variational Bayes diarization back-end (VBx) method, which clusters x-vectors using a Bayesian hidden Markov model (BHMM). The AHC in their experiments showed 3.96% DER, whereas VBx with a single Gaussian model per speaker showed 2.10% DER for the AMI Headset-Mix evaluation set. Since the analysis in both of the abovementioned papers focuses on oracle VAD, they can be comparable with our results, presented in Section 4.3.

In our research, we applied full-corpus-ASR [10] partitioning of meetings and used the evaluation part of the lapel-mix AMI corpus for diarization experiments and system fusion. For the training and evaluation of the speaker counter and overlapping speech detector models the training and evaluation parts of the Array1-01 AMI corpus were used, respectively. Additionally, the following datasets were used during the intermediate steps of our proposed approach.

The **Voxceleb1** dataset [23,24] is composed of audio files extracted from YouTube videos and contains 4874 utterances recorded at 16 kHz. The speakers span a wide range of different ethnicities, accents, professions and ages. Segments include interviews from red carpets, outdoor stadiums and indoor studios, speeches given to large audiences, excerpts from professionally-shot multimedia, and even crude videos shot on hand-held devices. Crucially, all are degraded with real-world noise, consisting of background chatter, laughter, overlapping speech, and room acoustics [25]. The quality of the recording equipment and channel noise quantity also vary quite noticeably.

The **NIST SRE 2019 evaluation** dataset [26] is composed of PSTN and VoIP data collected outside of North America, spoken in Tunisian Arabic, and contains 1364 enrollment and 13,587 test utterances recorded at 8 kHz. Speakers were encouraged to use different telephone instruments (e.g., cell phones, landlines) in a variety of settings (e.g., a noisy cafe, a quiet office) for their initiated calls [26]. Enrollment segments approximately contain 60 s of speech to build the model of the target speaker. The speech duration of the test segments is further uniformly sampled with lengths varying from approximately 10 s to 60 s.

The **16 kHz training set**. For this set we concatenated VoxCeleb1 and VoxCeleb2 (SLR47) [25] corpora. We used videos from VoxCeleb1 and VoxCeleb2, and concatenated all the corresponding audio files into one chunk. Augmented data were generated using the standard Kaldi augmentation recipe (reverberation, babble, music and noise) using the freely available MUSAN and simulated room impulse response (RIR). In total, the training dataset contains 833,840 recordings from 7205 speakers.

The **8 kHz training set**. For this set we used a wide variety of datasets, containing telephone and microphone data from private datasets and from those available online. The dataset includes Switchboard2 Phases 1, 2 and 3, Switchboard Cellular, Mixer 6 Speech, data from NIST SREs from 2004 through 2010 and 2018, concatenated VoxCeleb 1 and 2 data,

extended versions of the Russian speech subcorpus named RusTelecom v2 and the RusIVR corpus. RusTelecom is a private Russian speech corpus of telephone speech, collected by call centers in Russia. RusIVR is a private Russian speech corpus containing speech, collected in different scenarios, such as noisy microphones, telephone calls, recordings from distant recorders, etc. All files are sampled at 8 kHz. In order to increase the amount and diversity of the training data, augmentation using the standard Kaldi augmentation recipe (reverberation, babble, music and noise) was applied using the freely available MUSAN and simulated room impulse response (RIR) datasets. In total, this training dataset contains 1,679,541 recordings from 33,466 speakers.

3. Methods

In this section a detailed description of the main systems included in our diarization pipeline is provided. The first four subsections (VAD, SC, OSD and QE) describe the models used in various types of speech boundary detection and the fusion of these models. Section 3.5 is devoted to speaker extraction models and the methods used for their training.

3.1. Voice Activity Detection

This work proposes a fusion method of three different models, each of which shows different quality on the same evaluation subset of the AMI corpus. The first one is the voice activity detector (VAD), which is trained purposefully for the task of speech boundary detection on the AMI corpus. In our case, we use the method proposed in [7], which adapts the idea of using the U-Net architecture for segmentation from the spatial to the time domain. This architecture was originally introduced in [27], as a fairly precise method for object localization in microscopic images. U-Net is a convolutional architecture, that involves the idea of the deconvolution of small and deep image representation with many small layers into an image of the original size by applying the upsampling operation. In this study, we apply a reduced version of the original U-Net architecture, which is presented in Figure 1. Since the task of detecting speech activity is a task of segmentation in the time domain, we apply the combination of Dice and cross-entropy losses as the main loss-function in the VAD model training process [28].

The training process pipeline of the model for the AMI task consists of two stages:

1. Fitting on the main training set;
2. Adaptation on the AMI corpus.

During the first stage of training we use the concatenation of the NIST 2002/2008 speech recognition datasets and the RusTelecom corpus, described in Section 2.2. This data setup leads to a confident VAD quality of about a 10% equal error rate (EER) on different configurations of the model. Adaptation on the AMI corpus is described via the same training process as during the first stage, but using a smaller learning rate for the fitting of new data without the loss of already learned knowledge. In general, and also in our specific case, the adaptation process should last for a small amount of training iterations to prevent the overfitting on the new adaptation dataset and the reduction of the previous ability to detect speech in common.

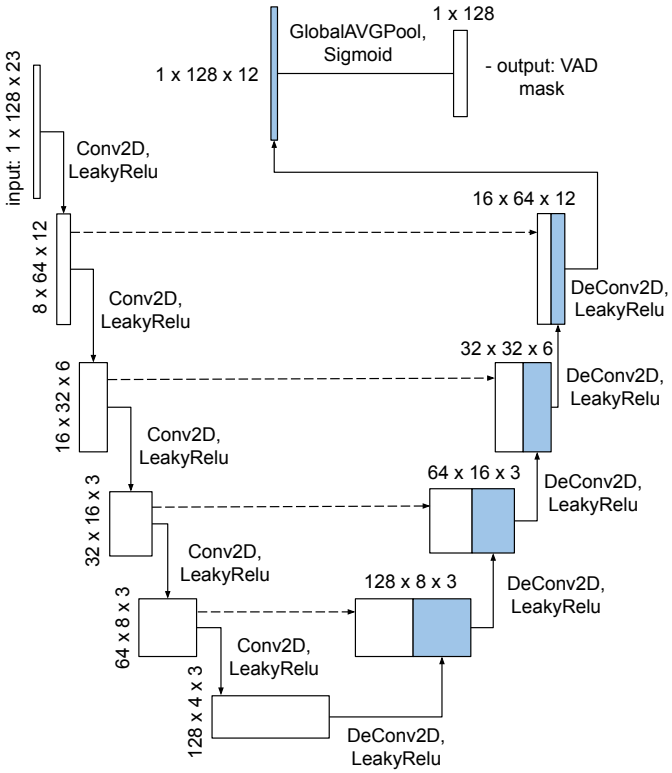

Figure 1. The applied U-Net VAD model architecture.

To compare VAD performance under the same conditions, we computed EER scores on the evaluation subset of the Third DIHARD Speech Diarization Challenge using our VAD and a similar method called SileroVAD [19]. The results were 16.9% EER using our method and 26.37% using SileroVAD.

3.2. Speaker Counter Model

Another model, the results of which can be interpreted as a time-domain speech detection markup, involves a detector of the number of simultaneous speakers, known as a speaker counter (SC).Theoretically, the estimation of the number of concurrent speakers is closely related to the problem of speaker identification, which is considered one of the tasks involved in speaker diarization [2,29–31].

We have compared the results of our model with the state-of-the-art systems in speaker number estimation presented in [11,32], which are based on convolutional neural networks. We have taken into consideration the models from the cited papers and the model presented in the current work, trained on 1-s recording segments to count the number of speakers. In [32], the authors present a model trained and evaluated on a synthetic dataset, which performs the speaker counting task with an average F1-score of 92.15% for classes with 0–3 speakers. In [11], the authors present a model trained and evaluated on mixtures of speaker recordings of the LibriSpeech dataset; this model achieves 77.3% accuracy for classes with 1–4 speakers.

Previously referenced works in the field mainly consider the synthetic mixtures of different speaker recordings and thus obtain permissible results for active speaker number estimations. We, however, focus specifically on real-life recordings of natural conversations that are contained in the AMI corpus, which reduces estimation quality. Our study has shown that no similar works in the field contain results based specifically on the AMI corpus; thus, exact comparison with our results is not possible. To train the model we applied data augmentation to the AMI corpus, and tested the model on the evaluation

set of this corpus. The model achieved an F1-score of 65.6% in real-speech conditions. The results are further discussed in Section 4.4.

The SC model solves the task of estimating the number of concurrent speakers, which is formulated as a classification task with 5 classes, from 0 to 4+ concurrent speakers. The model is based on the architecture described in [33]. The solution is based on the application of several deep neural network (DNN) models, a diagram of which is presented in Figure 2. These models consist of several convolutional layers, max-pooling layers, a Bi-LSTM layer with 40 hidden units for processing input features, and a fully connected layer, which implements the classification of the number of active speakers. The softmax function is used for the output layer; thus, the prediction is specified by the highest probability of the output distribution [34].

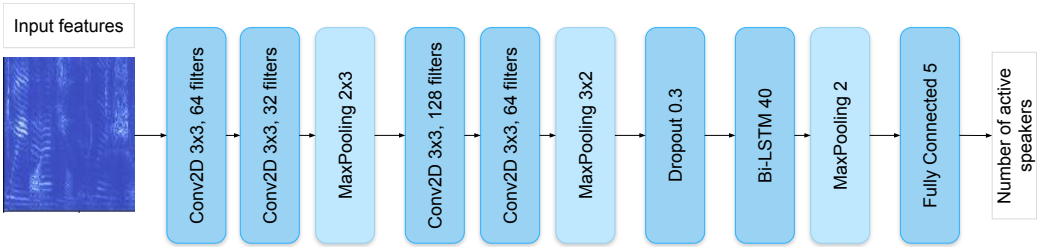

Figure 2. Speaker counter and overlapping speech detector model architecture.

To create the markup that is applicable to the SC model, we employed the manual annotations provided with the AMI corpus. These annotations specify the speech segments for each speaker. To obtain the markup with the number of active speakers, the beginning and end time stamps of all speech segments are combined into one set and sorted. This sorted set is then divided into subsegments, and for each subsegment the number of concurrent speakers is calculated using the original annotation [34]. Thus, we extract the subsegments where the number of simultaneous speakers remains unchanged, from 0 (silence) to 4+ overlapping speakers in a subsegment.

3.3. Overlapping Speech Detection

The model for the task of overlapping speech detection is similar to the SC model in its architecture (see Figure 2), but employs a different type of speech markup. The overlapping speech detector (OSD) is applied for the detection of overlapping speech segments in a conversation involving multiple speakers [11,35–40]. The OSD performs a binary classification task of detecting overlapping and non-overlapping speech. The markup for this task is produced using the speaker number markup of the SC task:

$$l_{OSD} = \begin{cases} 0, & l_{SC} \in \{0,1\}, \\ 1, & l_{SC} \in \{2,3,4+\}, \end{cases} \quad (1)$$

where l_{OSD} is an OSD segment class label and l_{SC} is an SC segment class label. The model presented in Figure 2 is trained with the OSD labels to perform the detection of overlaps. The model accepts the features extracted from short audio segments as input, whereas the output corresponds to the probabilities of the classes overlapping/non-overlapping per each speech segment.

3.4. QE-Vectors System

As an alternative method of separating speech and non-speech segments, we used the information about the acoustic characteristics of the signal from the microphone array of the AMI corpus. During evaluation, for each 0.5 s signal frame we estimated the SNR and RT60 parameters using the signal quality estimation (QE) model described in [9]. SNR and RT60 parameters were predicted for all 8 channels of the microphone array, so for each 0.5 s frame we obtained a vector of 16 values. The resulting vectors were clustered using the

K-means method into two clusters. To decide which of the clusters represented the speech segments and which represented non-speech, the centers of each cluster were compared with the mean speech vector, which is calculated in advance for the speech segments of the development part of the AMI database. This way it was possible to retrieve the markup of speech and non-speech segments for further fusing with a more accurate method, such as base DNN-VAD.

3.5. Speaker Extractor Models

This section contains the descriptions of the foremost architectural details, training techniques, and distinctive features of the implemented speaker verification models. The most common approaches for speaker verification today are based on blocks with one-dimensional (TDNN) or two-dimensional (ResNet) convolutions, which influenced our choice regarding the speaker extractors' architecture. Thus, in this paper, we explore the ECAPA-TDNN architecture, which develops the idea of an x-vector TDNN, and ResNet-based models, trained on 8 kHz and 16 kHz datasets. We used ResNet34 as a baseline model and attempted to increase the quality of the speaker model by increasing the depth and width of the layers and by investigating the model architectural configuration.

ECAPA-TDNN. Emphasized channel attention, propagation and aggregation in TDNN (ECAPA-TDNN), proposed in [13], is a modification of the standard time delay neural network (TDNN) architecture, containing squeeze-excitation (SE) blocks and Res2Net modules in the frame level with hierarchical filters for the extraction of features of different scales. To process signals of arbitrary duration, the architecture uses attentive statistic pooling instead of the usual statistic pooling. For this work, we chose the official pre-trained model from https://github.com/speechbrain/speechbrain (accessed on 2 March 2021), which was trained on the original VoxCeleb1 and VoxCeleb2 datasets with SpecAugment [41], speed perturbation, noise, and reverberation augmentation. In detail, the model configuration and training are described in [42].

ResNet34. This extractor is based on the ResNet34 model with some modifications, as well as the Maxout activation function on the embedding layer, set to one stride in the first BasicBlock and changed to a simple Conv2D stem block. This model was trained on the 16 kHz dataset with local CMN-normalization and global CMVN-normalization, sequentially. During the training process, the extractor handles short audio segments with the length fixed at 2 s using AM-Softmax loss. During the training stage, the parameters m and s were set equal to 0.2 and 30, respectively. The learning rate was set to 0.001 for the first two epochs, and then it was decreased by a factor of 10 for each consecutive epoch. Detailed information about the model configuration is presented in [7].

ResNet72. For this extractor, the ResNet architecture from [43] was adapted to the task of speaker verification. The model was trained on the 16-kHz dataset with local CMN-normalization. As a loss function, we used adaptive curriculum learning [44]. Parameters m and s were respectively equal to 0.35 and 32 during the whole training stage. The optimization process was based on the SGD optimizer with the momentum of 0.9 and weight decay of 10–5 for the head and 10–6 for the rest of the layers. Moreover, for learning rate control the OneCycleLR scheduler was used with the maximum learning rate fixed to 2. To increase the training speed, we also used AMP (Automatic Mixed Precision) with half-precision, which raised the batch size per GPU. The model was trained during 15 epochs on randomly sampled 4-s crops of each utterance from the training dataset. The comprehensive description of the model configuration is presented in [45].

ResNet101_8k. The ResNet101 model modification included the Maxout activation function on the embedding layer, set to one stride in the first BotleneckBlock and changed to a simple Conv2D stem block, which provided the basis for this extractor. The training set for this model was the 8 kHz dataset with local CMN-normalization performed in several stages. In the first stage, the model was trained for 20 epochs on randomly sampled 6-s crops of each training dataset utterance. AAM-Softmax losses with parameters m and s were respectively equal to 0.35 and 32. In the second stage, crop duration was increased to

12 s and the parameters of AAM-Softmax loss were set to 0.45 and 32, respectively. Then, the model went through 10 epochs of training with these settings.

ResNet101_48_8k. The ResNet101_8k model was set to be used directly as a basis for the extractor at hand, with the extension number of convolution layers in the model architecture ranging from 32 N to 48 N, where $N \in \{1, 2, 4, 8\}$ for different model blocks. Higher quality was achieved through an increase in the size of convolutional layers; however, the increase in the number of model parameters resulted in an increase of training time compared to the ResNet101_8k model.

3.6. Model Fusion

In order to improve the accuracy of speech boundary detection, we conduct a series of experiments by fusing the speech markups obtained using DNN-VAD with the markups obtained using other systems discussed above: SC, OD and QE. The following options for fusion were considered:

- VAD + QE, on the assumption that information about acoustic conditions (RT60, SNR) can correlate with the speech component properties and thereby improve VAD markup;
- VAD + SC, on the assumption that the above-described model for counting the number of speakers can be considered as an alternative approach to the detection of speech activity;
- VAD + QE + SC + OD, on the assumption that the SC + OD fusion is shown to produce good quality estimates (see Section 4.4), and the combination of different approaches in the conditions of a multispeaker conversation can improve the accuracy of voice annotation compared to baseline DNN-VAD.

According to the diarization pipeline, the obtained speech markup was then used for speaker model extraction; so, the accuracy of speech boundary detection affected the overall diarization quality. We describe the details of the applied voice annotation system fusion and analyze the influence of different fusion approaches on speaker diarization quality in terms of DER in Section 4.5.

4. Experiments

Our main experiment was aimed at the fusion of different speech analytics techniques to improve the overall quality of diarization. A description is available in Section 4.5, which includes the comparison of different combinations of speech markup methods in terms of DER: VAD, QE, SC and OSD, outlined above. In the final experiment we intended to apply the speaker extractor model chosen in Sections 4.1 and 4.2, the selected clustering method (Section 4.3), and the SC + OSD fusion method (Section 4.4) in combination.

Thus, Sections 4.1–4.4 describe preliminary experiments aimed at independently improving each of the stages of fusion. Section 4.1 contains the comparison of different speaker extractor models in the speaker verification task. We tested the generalizability of different neural network architectures trained on the 16 kHz training set and the combined 8 kHz training set described in Section 2.2 to cope with unknown evaluation data. Section 4.2 compares the same speaker extractor models for an AMI-based diarization problem using ideal VAD markup. As a result of these experiments, we defined the ResNet101_8k model as the most optimal, so it was chosen for the final experiments. Section 4.3 is devoted to the next stage of the diarization pipeline—the comparison of clustering methods for extracted voice models of speakers. The experiments were carried out on the ResNet101_8k and ECAPA_TDNN models. Finally, Section 4.4 focuses on SC and OSD fusion to refine speech boundaries under interruption conditions.

4.1. Speaker Verification

We investigated the quality of different speaker extractors for the speaker verification task on the Voxceleb1 test and NIST SRE 2019 eval sets. Since the speaker verification system receives two sets of speech recordings during the evaluation process, originating

either from the same speaker or from two different speakers, there arise two types of errors: the false acceptance rate (FA) and the false rejection rate (FR), which are dependent on the decision threshold. Then, the equal error rate (EER) is the point at which both rates are equal to each other. The lower the EER value, the higher the overall accuracy of the biometric system [46].

In order to take into account the different FR and FA error costs, there exists the detection cost function (DCF or C_{det}) measure, which is described by the following equation:

$$C_{det}(\theta) = C_{FR} \cdot P_{tar} \cdot P_{FR}(\theta) + C_{FA} \cdot (1 - P_{tar}) \cdot P_{FA}(\theta), \quad (2)$$

where C_{FR} and C_{FA} are the estimated cost parameters for false accept and false reject errors, P_{tar} is a prior probability of targets in the considered application; error-rates P_{FR} and P_{FA} are determined by counting errors during evaluation, and θ is the decision threshold.

The optimal value of DCF obtained via the adjustment of the detection threshold is minDCF. In our experiments we used the EER measure and minDCF with the *a priori* probability of the specified target speaker (P_{tar}) set to 0.01, $C_{FR} = 1$, $C_{FA} = 1$.

The main idea of this experiment consisted of comparing the models trained on different types of data (8 kHz or 16 kHz) and tested on the test subsets of the same data (8 kHz or 16 kHz as well). For this task, we upsampled the 8 kHz audio files to 16 kHz or downsampled the 16 kHz audio files to 8 kHz. For ECAPA-TDNN, we used the official implementation of the model and data processing pipeline from [42]. Note that VAD was not used for feature processing for ECAPA-TDNN, according to the official implementation. No normalization or adaptation techniques were used for speaker embedding comparisons. Simple cosine similarity between speaker embedding vectors was used as a score for speaker model comparisons. The results of the comparison between the speaker verification models are presented in Table 2. To determine the confidence intervals for the EER estimates, we used the fast equal error rate confidence interval (FEERCI) algorithm to calculate non-parametric, bootstrapped EER confidence intervals [47]. The main idea of this approach is to use random subsets from genuine and imposter score lists for estimation. We used 95% confidence intervals on 50 bootstrap iterations for this task. We estimated the confidence intervals for the fixed architecture of the model and its weights due to the significant computational complexity of the task of retraining the speaker recognition model. Based on the results, the following conclusions can be drawn:

- The system trained for specific data types works better on the test set of the corresponding type. Data type mismatches led to significant quality degradations for all tested systems;
- The quality degradation of models trained on a combined dataset (ResNet101_8k, ResNet101_48_8k) was less than the degradation of the models trained only on VoxCeleb datasets. This can be explained by the growth of the generalizing ability of the network with the increase in samples of the training dataset;
- The ResNet-based models trained on VoxCeleb datasets showed better quality on out-of-domain tasks, compared to ECAPA-TDNN, which can be observed by comparing the results of speaker verification. However, ECAPA-TDNN shows a better result on the in-domain test dataset.

Table 2. Results of speaker verification systems for the in-domain Voxceleb1 test and the out-of-domain NIST SRE 2019 eval sets in terms of the EER ± 95% confidence interval/minDCF (%). Lower error values are better.

Model	EER ± ci = 0.95/MinDCF0.01 (%)		
	Train Set	VoxCeleb1 Test	NIST SRE 2019 Eval Set
ECAPA-TDNN	VoxCeleb1, VoxCeleb2 [13]	0.71 ± 0.09/0.092	14.52 ± 0.11/0.785
ResNet34	16 kHz	1.18 ± 0.10/0.126	14.37 ± 0.12/0.748
ResNet72	16 kHz	1.11 ± 0.07/0.093	12.75 ± 0.11/0.723
ResNet101_8k	8 kHz Combined	1.54 ± 0.15/0.156	2.97 ± 0.07/0.276
ResNet101_48_8k	8 kHz Combined	1.43 ± 0.14/0.135	2.85 ± 0.07/0.280

4.2. Diarization

The quality of various speaker verification systems was compared during the task of diarization. The embeddings of each continuous speech segment were extracted with the chosen sliding window and shift duration values. We investigated the influence of these parameters based on the results of diarization in terms of the diarization error rate (DER), which consists of three types of error: speaker error, missed speech, and false alarm. DER is denoted as

$$\mathrm{DER} = E_{spkr} + E_{miss} + E_{fa}, \tag{3}$$

where E_{spkr} is the speaker error, the percentage of the scored time when a speaker ID is assigned to the wrong speaker. This type of error does not account for overlapping speakers or any error situated within non-speech frames. E_{miss} is missed speech—the percentage of scored time when a hypothesized non-speech segment corresponds to a reference speaker segment. E_{fa} is false-alarm speech, the percentage of scored time when a hypothesized speaker is labeled as non-speech in the reference annotation [48].

We used the DER metric configured according to NIST: a forgiveness collar of 0.25 s was used and the speaker overlap regions were ignored during scoring. We used the ideal VAD markup computed from the ground truth information from the AMI dataset. The results of the comparison of the speaker verification models in terms of DER are presented in Table 3. The development set was used for tuning the spectral clustering parameters. The tuned parameters were used to perform the diarization task on the evaluation set.

Table 3. Results of investigated systems for AMI Development and Evaluation sets (dev/eval) in terms of DER (%) depending on the set sliding window and shift duration values. Lower error values are better. Upper and lower 0.95 confidence intervals are indicated with a dash.

Model	DER (%) on dev with ci = 0.95			
	Win = 1.0 s Shift = 0.5 s	**Win = 1.5 s Shift = 0.75 s**	**Win = 2.0 s Shift = 1.0 s**	**Win = 3.0 s Shift = 1.5 s**
ECAPA-TDNN	3.90–5.29–6.14	1.95–2.50–2.84	1.65–2.17–2.46	1.92–2.37–2.65
ResNet34	3.51–5.12–6.09	1.48–2.35–2.76	1.54–**1.78**–2.03	1.74–2.65–3.05
ResNet72	2.48–**3.39**–3.96	1.35–2.10–2.44	1.38–1.91–2.21	1.72–**2.12**–2.38
ResNet101_8k	2.44–**3.38**–3.89	1.46–2.12–2.43	1.30–1.90–2.15	1.76–2.28–2.56
ResNet101_48_8k	2.56–3.99–4.72	1.40–**1.88**–2.16	1.42–**1.80**–2.04	1.82-**2.15**-2.43
Model	**DER (%) on eval with ci = 0.95**			
	win = 1.0 s shift = 0.5 s	**win = 1.5 s shift = 0.75 s**	**win = 2.0 s shift = 1.0 s**	**win = 3.0 s shift = 1.5 s**
ECAPA-TDNN	3.44–3.99–4.52	2.45–2.81–3.15	2.15–2.42–2.67	2.52–2.81–3.10
ResNet34	2.93–3.73–4.39	1.93–2.46–2.91	2.00–3.20–3.85	2.05–2.33–2.78
ResNet72	2.92–**3.59**–4.32	1.77–1.94–2.30	1.41–**1.58**–1.85	2.02–2.56–3.05
ResNet101_8k	3.52–3.90–4.61	1.85–2.02–2.36	1.62–1.81–2.10	1.99–**2.17**–2.53
ResNet101_48_8k	2.97–3.66–4.19	1.63–**1.75**–1.95	1.47–1.64–1.82	2.00–2.23–2.52

To determine the confidence intervals for the DER estimates we used the following algorithm:

1. Input: $(\{w_i, \ldots, w_n\})$; w—diarization results for each file in subset, n—number of unique files in subset;
2. Choose random subsets of size $n-2$ from the entire subset and compute the DER metric for the files in each of $n-2$ size subsets. We use $n-2$ files to maximize the size of subset, while assuring a sufficient variability of subsets;
3. Repeat the 2nd step 50 times, sort all computed DER results, compute mean and 95% confidence interval thresholds.

Based on the presented results, the following conclusions were drawn:

- The quality of the ResNet-based model outperformed ECAPA-TDNN in the diarization task for the AMI dataset;
- The best quality on the evaluation set of AMI was achieved using the sliding window and shift duration values set to 2.0 s and 1.0 s. respectively;
- High-quality diarization was achieved using the model trained on the 8 kHz dataset, indicating that the data variability of the training dataset was no less important for achieving better diarization results than the type of train data. The quality of the model trained on the 8 kHz dataset matched or even exceeded the quality of models trained on test-like data;
- The sizes of the development and evaluation parts of the AMI dataset were probably insufficient for quality assessments with high confidence intervals. For example, the quality of all systems on the evaluation set for sliding window and a shift duration values of 1.5 s and 0.75 s, accordingly, were comparable in the confidence interval. Furthermore, the quality of ResNet72, ResNet101_8k and ResNet101_48_8k models for the window and shift parameters larger than the abovementioned values were comparable.

Here and in other tables of this paper, some metrics contains range values which correspond to boundaries of confidence intervals. These values are separated with a minus sign ("−").

4.3. Clustering

This section compares the clustering algorithm proposed in [17,21] with other clustering methods from Kaldi and sklearn (https://scikit-learn.org/stable/ accessed on 18 May 2021) libraries. For this set of experiments, we used the ResNet101_8k speaker verification system with the best configuration for this model, as discussed in Section 4.2. First, we applied clustering methods with a reference number of clusters, where the number of speakers was computed based on original AMI annotation.

1. Spectral Clustering with cosine affinity and unnormalized modification;
2. Spectral Clustering with cosine affinity and without unnormalized modification from the sklearn package;
3. The k-means clustering algorithm from the sklearn package;
4. Gaussian mixture model with tied covariance matrices, trained on speaker-embedding vectors. The number of used gaussians was chosen according to the reference number of speakers;
5. The density-based spatial clustering of applications with noise (DBSCAN) algorithm was tested but a stable configuration yielding sufficient quality was not achieved; thus, the results of the method at hand are not presented in Table 4.

Second, several clustering algorithms with automatic estimation of the number of clusters were compared.

1. Spectral clustering with cosine affinity and unnormalized modification. The number of speakers was estimated using the maximum eigengap between eigenvalues, computed on the Laplacian matrix, which was calculated based on the pruned cosine similarity matrix between speaker-embedding vectors [21].

2. Agglomerative clustering with cosine affinity and average linkage. The sklearn and Kaldi implementations of this clusterization were used, with insignificant differences between the results in terms of DER.

All the results achieved for the clustering methods are presented in Table 4. Based on these results, better performance was achieved using an unnormalized modification of the spectral clustering algorithm for both cases of the specified reference number of speaker clusters and for the estimated number of clusters.

For the visualization of the markups produced by different speaker models, we used t-distributed stochastic neighbor embedding (TSNE). TSNE results for the ResNet101_8k and the ECAPA-TDNN models are presented in Figures 3 and 4, respectively. The original speaker embeddings and their decomposed representations after spectral decomposition performed internally by SC are visualized. We used the ideal clustering markup with removed speech segments of overlapping speech, estimated after K-means testing for the original speaker embeddings after SC clustering. Note that for both models the results of DER for this file consisted of approximately 1–2% of samples.

Table 4. Results of comparing the clustering algorithms for the AMI Development and Evaluation sets (dev/eval) in terms of DER (%) for the reference and the estimated number of speakers in each single recording. The methods are numbered according to their presentation in Section 4.3. Lower error values are better.

Clustering	Reference Speakers	Dev	Eval
1. Spectral Clustering unnorm	True	1.91	1.81
2. Spectral Clustering	True	2.18	3.05
3. Kmeans	True	2.31	4.38
4. GaussianMixture	True	2.39	4.79
1. Spectral Clustering unnorm	False	2.13	2.78
2. Agglomerative Clustering	False	3.64	3.83

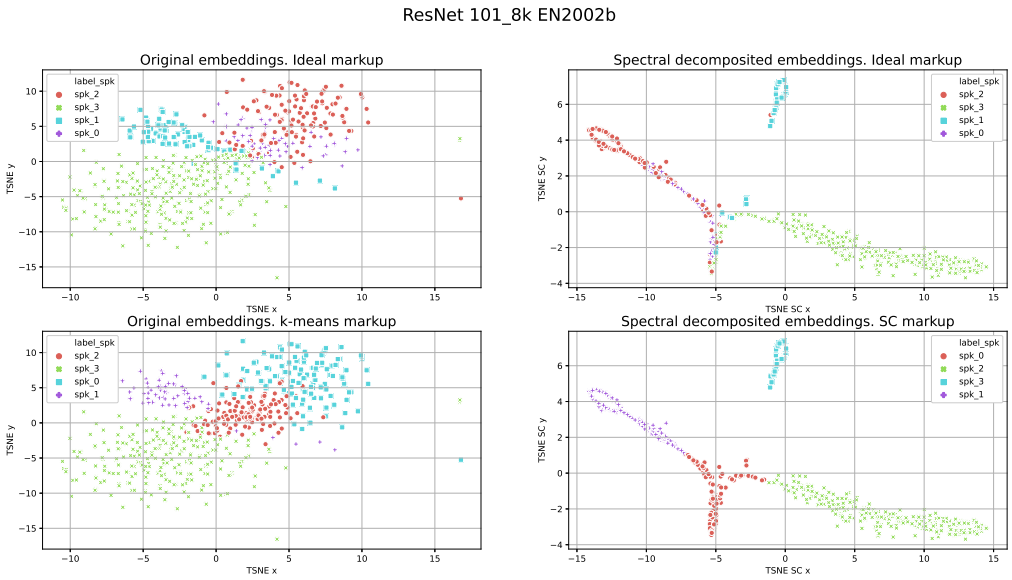

Figure 3. Results of TSNE visualization computed by means of the ResNet101_8k model and EN2002b lapel-mix file from the AMI evaluation dataset. All overlapping speech segments were removedto reduce uncertainty. In the **left** parts of the images, the original speaker embeddings are used; in the **right** parts the speaker embeddings after spectral decomposition are used.

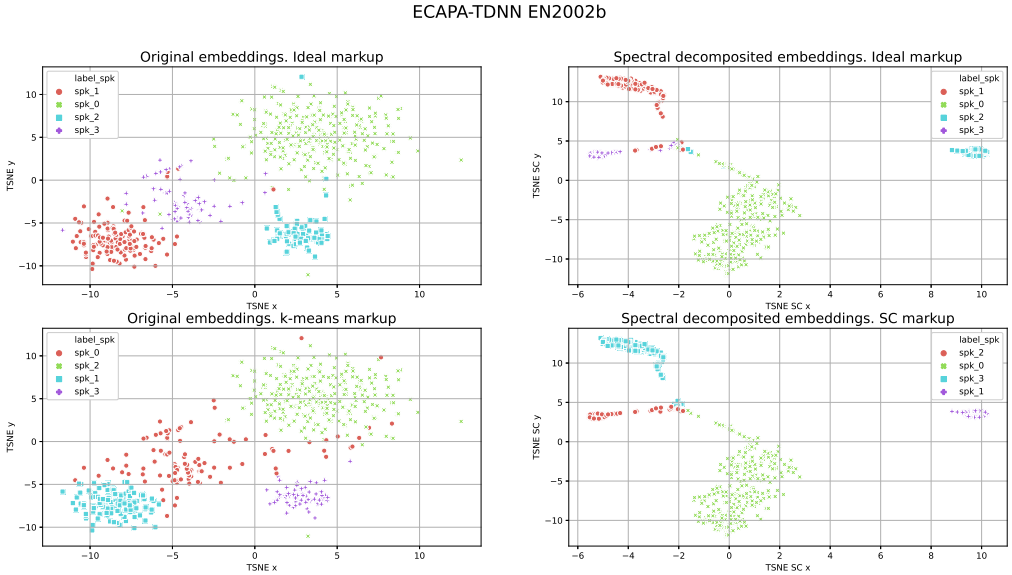

Figure 4. Results of TSNE visualization computed by means of the ECAPA-TDNN model and EN2002b lapel-mix file from the AMI evaluation dataset. All overlapping speech segments were removed to reduce uncertainty. In the **left** parts of the images, the original speaker embeddings are used; in the **right** parts the speaker embeddings after spectral decomposition are used.

4.4. Fusion of SC and OSD Models

The fusion of the SC and OSD models was proposed to improve the quality of diarization during simultaneous speech. Combining these models, the SC performance can potentially be improved at the points of overlapping speech (Figure 5). In a case in which the SC model does not perform well for the speaker number classes of 2, 3 and 4+ simultaneous speakers, but the binary problem of overlap/non-overlap detection yields sufficient estimation quality, the performance of SC may be tuned according to OSD estimates [34]. If SC underestimates the number of speakers, marking most samples as one speaker, then, by fusing SC with OSD the samples with one speaker can be excluded, eliminating the classification error for samples with more than two reference speakers estimated as one speaker. On the other hand, SC may overestimate the number of speakers. In this case, by fusing the models and excluding samples containing 2–4+ speakers, the estimation quality for zero and one speakers can be improved [34].

The principle of the proposed fusion method lies in adjusting the SC class probabilities based on the decision of OSD [34]. If the decisions of the two models are logically in concurrence with one another, the SC probabilities are increased by applying a weight coefficient α. On the other hand, if the decisions of the two models are not in concurrence, the SC probabilities are reduced. Specifically, if OSD defines a speech segment as non-overlapping, then this estimate is used to increase SC probabilities for labels 0 and 1 (no active speaker, one active speaker), and reduce the probabilities for labels 2, 3, and 4+ (2, 3, 4+ active speakers). For segments with estimated overlapping speech, the SC probabilities for labels 2, 3, 4+ are increased, and reduced for labels 0 and 1. The fusion process is generalized by means of the following equation [34]:

$$P_{fusion} = \begin{cases} P_{SC}(l_{SC}) + \alpha \cdot (1 - P_{SC}(l_{SC})), & \hat{l}_{OSD} = 0, \hat{l}_{SC} \in \{0,1\}, \\ P_{SC}(l_{SC}) - \alpha \cdot P_{SC}(l_{SC}), & \hat{l}_{OSD} = 1, \hat{l}_{SC} \in \{0,1\}, \\ P_{SC}(l_{SC}) - \alpha \cdot P_{SC}(l_{SC}), & \hat{l}_{OSD} = 0, \hat{l}_{SC} \in \{2,3,4+\}, \\ P_{SC}(l_{SC}) + \alpha \cdot (1 - P_{SC}(l_{SC})), & \hat{l}_{OSD} = 1, \hat{l}_{SC} \in \{2,3,4+\}, \end{cases} \quad (4)$$

where $P_{SC}(l_{SC})$ are the probabilities of l_{SC} labels for a given segment, $\hat{l}_{OSD}$ and $\hat{l}_{SC}$ are the OSD and SC estimates, respectively, and α is the weight coefficient. The value for α was chosen to be equal to 0.5 during our fusion experiments.

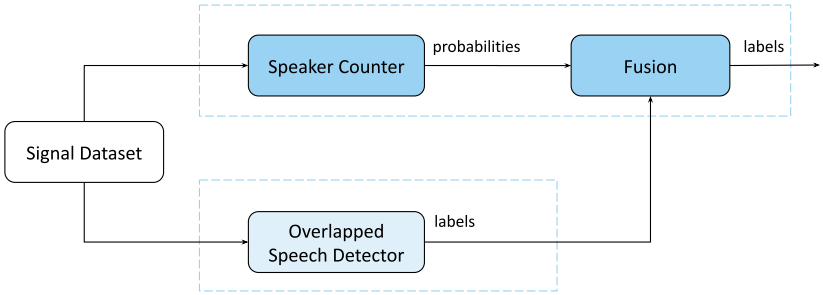

Figure 5. Diagram of speaker counter and overlapping speech detector fusion.

Table 5 presents the results for each class, as well as their weighted average in the metrics of accuracy, completeness, and F1-score. We used the following equation to calculate the F1-score metric, which combines the values of precision and recall; the equation was applied as in the Python module scikit-learn (https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html accessed on 14 June 2021):

$$\text{F1-score} = 2 \cdot \frac{precision \cdot recall}{precision + recall}. \quad (5)$$

Table 5 also presents the confidence intervals for the F1-score. To calculate the confidence interval, we take the model results calculated on all files, randomly select 80% of labels, calculate the F1-score for them, and repeat this procedure 50 times. Then we compute 95% confidence intervals by applying the Python module scipy.stats (https://docs.scipy.org/doc/scipy/reference/stats.html accessed on 21 June 2021), taking into account the fact that the obtained F1-score values belong to a normal distribution.

The SC model fused with OSD yields better speaker number estimation quality compared to the non-fused SC model. Experimental results show an increase in recognition quality for all classes, in particular for labels 1 and 2. The recall metric shows no increase, probably due to the fact that SC is initially efficient in classifying segments with numbers of speakers larger than two. The recall increases only when the number of speakers is equal to 1. This means that SC often tends to overestimate the number of speakers. A noticeable increase occurs in the precision metric. This indicates that SC has become more efficient in cases where the number of speakers is equal to 1, whereas previously it tended to overestimate the number of speakers in this case.

Table 5. Speaker counter and overlapping speech detector fusion results.

Method	Class	Precision	Recall	F1-Score ± ci = 0.95
SC	0	0.621	0.755	0.682 ± 0.003
	1	0.736	0.593	0.657 ± 0.001
	2	0.274	0.390	0.322 ± 0.002
	3	0.196	0.242	0.216 ± 0.003
	4+	0.136	0.097	0.113 ± 0.008
	weighted	0.593	0.546	0.562 ± 0.001
SC + OSD	0	0.639	0.744	0.688 ± 0.002
	1	0.781	0.772	0.777 ± 0.001
	2	0.400	0.412	0.406 ± 0.002
	3	0.295	0.222	0.253 ± 0.004
	4+	0.163	0.093	0.119 ± 0.008
	weighted	0.654	0.659	0.656 ± 0.001

4.5. Fusion of Voice Annotation Methods

In the following experiments, we investigated the system VAD markup and reference number of speakers used for diarization. We focused on ways to improve the speech markup by fusing the results of various algorithms presented earlier in this study. For this purpose, a better configuration of VAD was obtained in terms of DER for the AMI evaluation dataset. We used results estimated by applying only the VAD markup as a baseline for other fusion results. We used the ResNet101_8k speaker verification system with the best configuration for the construction of diarization vectors using the system VAD markup. For each segment of the source utterance, the fusion algorithm computed a simple weighted sum of the probability of the presence of speech in the segment according to Equation (6), where p_1 corresponds to the probability of definition of an utterance segment as speech by VAD, p_2 corresponds to the probability of definition of the utterance segment as speech by another method, and α serves as a weight coefficient. Note that the coefficient α in this regard differs from the coefficient applied in SC + OSD fusion in Section 4.4. The best configuration of the independent fusion of VAD with QE and SC with OSD for the case when more than two methods are used is denoted as

$$markup_{fusion} = \begin{cases} 1, \alpha \cdot p_1 + (1 - \alpha) \cdot p_2 > 0.5, \\ 0, \text{else}. \end{cases} \quad (6)$$

Note that the better VAD configuration is slightly different for the development and evaluation sets of AMI. The experimental results of fusion are presented in Table 6. The weight α differs depending on the methods to which fusion is applied. This allows one to estimate the best values of α, which are presented in the table as well. Based on these results, the following conclusions can be drawn:

- Using the fused VAD significantly reduces the diarization error compared the baseline VAD system. In cases of the diarization of utterances recorded in simple environments (e.g., low noise and reverberation levels), it can be assumed that VAD error will prevail over speaker errors in terms of DER.
- The QE and SC algorithms can be used for the clarification of FR errors, which leads to an error decrease in terms of DER. This effect can be observed for the high values of the fusion weight coefficient, when QE and SC help to clarify the markup in case of VAD uncertainty.
- The application of all the algorithms considered in the article in concurrence leads to a significant improvement in the quality of diarization and reduces both FR and FA errors. This effect can be observed for small α values, and probably indicates that VAD can be replaced by a fusion of the QE, SC and OSD methods.

Table 6. Results for the AMI evaluation set in terms of DER decomposed to false rejection errors, false acceptance errors and speaker errors for the fusion of different model combinations.

Fusion	DER	FR	FA	SE
VAD (baseline)	39.91–40.35–40.83	19.58	12.6	8.17
VAD + QE, α = 0.95	28.23–28.68–29.12	4.94	15.04	8.70
VAD + SC, α = 0.58	26.80–27.28–27.71	6.38	14.37	6.54
VAD + QE + SC + OSD, α = 0.05	13.76–14.31–14.90	1.46	8.03	4.82

The computation of the confidence interval for the DER metric was inspired by the Python scipy.stats.bootstrap method, and consisted of the following steps:

1. Create samples of length $N - 2$ from the set of AMI scenarios, where N is the number of scenarios;
2. Compute the overall DER on that extracted subset;
3. Repeat steps 1. and 2. M times. In our case $M = 100$.

4. Compute 0.05 and 0.95 quantiles for the obtained array. The two resulting numbers correspond to the lower and upper bounds of the confidence interval.

5. Discussion

The performance of various speaker verification systems were investigated in this work with the aim of improving diarization quality. We found that the ResNet model outperformed the ECAPA-TDNN model, despite achieving worse quality in the in-domain task compared to the AMI speaker verification task. The best model configuration, diarization setup, and clustering methods were obtained for the AMI dataset to obtain better diarization quality in terms of DER. As far as the authors know, we report state-of-the-art results for the AMI dataset, reaching 1.58% DER for the AMI evaluation set for the lapel-mix microphone using a deeper ResNet model with better speaker discrimination ability. During our investigation we found that the data variability of the training dataset is important for the achievement of better diarization results, even while using 16 kHz telephone data. Further improving the quality of speaker verification models seems to be a good way to improve the quality of diarization. We confirmed the best quality of clustering for the unnormalized modification of spectral clustering compared to all other investigated methods of clustering.

In the case of the use of VAD markup, some ways to fuse different methods allowing an improvement in the quality of the estimated speech markup were considered. The fusion of VAD with methods of quality estimation, the speaker counter, and the overlapping speech detector allowed us to significantly decrease DER values from 40.35% DER to 14.31% DER. This improvement was achieved by reducing the number of errors of false rejection of speech fragments, as well as clarifying the markup of speech fragments by reducing the number of errors of false acceptance. At the same time, the number of errors involving the mixing up of speakers was still quite high, which was probably due to the difficulties in identifying mixed speech. Finding solutions for the diarization of mixed speech remains an important challenge.

By fusing the SC and OSD methods, the performance of speaker number estimation improved from 0.562 (by applying only SC) to 0.656 in terms of the F1-score. However, the estimation quality was less if the unbalanced real-life data from the AMI corpus was used for training, compared to the results obtained on synthetic mixtures of speakers. This is quite expected an explainable due to the nature of real conversations, specifically acquired from the far-field. It has been noted, though, that data augmentation which improves the balance of classes increases the estimation quality of the model. Therefore, further steps in our research include improving the fused SC and OSD system by training it on a more balanced and diverse dataset.

6. Conclusions

In this paper, various methods of audio signal processing, such as speaker number estimation, overlapping speech detection, quality estimation, and different speaker verification models, were considered with the aim of improving the quality of diarization algorithms in terms of DER. Future developments in the considered direction of research include improving the speaker verification models and training more robust methods on speech segment markup, which will allow us to further improve the quality of diarization. Furthermore, training end-to-end models for speaker diarization, employing the studied principles of speaker information estimation, may prove to be advantageous.

Author Contributions: Conceptualization, S.A. and A.G.; methodology, S.A., A.G., M.V., and A.L.; software, A.G. and A.L.; validation, S.A., A.G., and A.L.; formal analysis, S.A., A.G. and A.L.; investigation, A.G., M.V., A.L., V.Z., V.K. (Vlada Kapranova), E.T., and E.E.; resources, A.G., M.V., A.L., V.Z., V.K. (Vlada Kapranova), E.T., and E.E.; data curation, A.G., M.V., A.L., V.Z., V.K. (Vlada Kapranova), E.T., and E.E.; writing—original draft preparation, A.G., M.V., and A.L. with significant contributions from V.Z., V.K. (Vlada Kapranova), E.T., and E.E.; writing—review and editing, S.A., V.K. (Vladimir Kabarov), and Y.M.; visualization, A.G., A.L., and E.E.; supervision, V.K. (Vladimir

Kabarov) and Y.M.; project administration, S.A.; funding acquisition, S.A. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially financially supported by ITMO University.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data applied in this study were obtained from: the AMI corpus (https://groups.inf.ed.ac.uk/ami/corpus/ accessed on 30 May 2021); Voxceleb1 and Voxceleb2 corpora (https://www.robots.ox.ac.uk/vgg/data/voxceleb/ accessed on 15 May 2021); the NIST SRE 2019 evaluation dataset from the NIST 2019 Speaker Recognition Evaluation challenge (https://www.nist.gov/itl/iad/mig/nist-2019-speaker-recognition-evaluation accessed on 12 March 2021). Access restrictions are applied to the RusTelecom and RusIVR private Speech Technology Center corpora.

Conflicts of Interest: Authors Sergei Astapov, Aleksei Gusev, Marina Volkova, Aleksei Logunov, Valeriia Zaluskaia, Vlada Kapranova, and Yuri Matveev were employed by the company STC-innovations Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

VAD	Voice Activity Detector
SC	Speaker Counter
OSD	Overlap Speech Detector
QE	Quality Estimation
DNN	Deep Neural Network
TDNN	Time Delay Neural Network
ECAPA-TDNN	Emphasized Channel Attention, Propagation, and Aggregationin TDNN
RIR	Room Impulse Response
RT60	Reverberation Time (the time of sound pressure reduction by 60 dB)
SNR	Signal to Noise Ratio
STFT	Short-Term Fourier transform
LMFB	Log Mel-filter Bank Energies (features)
MFCC	Mel Frequency Cepstral Coefficients (features)
CMN	Cepstral Mean Normalization
CMVN	Cepstral Mean and Variance Normalization
EER	Equal Error Rate
DER	Diarization Error Rate
SGD	Stochastic Gradient Descent
GPU	Graphics Processing Unit
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
TSNE	t-Distributed Stochastic Neighbor Embedding
AM-Softmax	Additive Margin Softmax (loss)
AAM-Softmax	Additive Angular Margin Softmax (loss)
AMP	Automatic Mixed Precision

References

1. Laptev, A.; Andrusenko, A.; Podluzhny, I.; Mitrofanov, A.; Medennikov, I.; Matveev, Y. Dynamic acoustic unit augmentation with bpe-dropout for low-resource end-to-end speech recognition. *Sensors* **2021**, *21*, 3063. [CrossRef]
2. Tranter, S.E.; Reynolds, D.A. An overview of automatic speaker diarization systems. *IEEE Trans. Audio Speech Lang. Process.* **2006**, *14*, 1557–1565. [CrossRef]
3. Yella, S.H.; Bourlard, H. Overlapping Speech Detection Using Long-Term Conversational Features for Speaker Diarization in Meeting Room Conversations. *IEEE Trans. Audio Speech Lang. Process.* **2014**, *22*, 1688–1700. [CrossRef]

4. Medennikov, I.; Korenevsky, M.; Prisyach, T.; Khokhlov, Y.Y.; Korenevskaya, M.; Sorokin, I.; Timofeeva, T.; Mitrofanov, A.; Andrusenko, A.; Podluzhny, I.; et al. Target-speaker voice activity detection: A novel approach for multi-speaker diarization in a dinner party scenario. In Proceedings of the Interspeech 2020, Shanghai, China, 25–29 October 2020.
5. Dehak, N.; Kenny, P.; Dehak, R.; Dumouchel, P.; Ouellet, P. Front-end factor analysis for speaker verification. *IEEE/Trans. Audio Speech Lang. Process.* **2011**, *19*, 788–798. [CrossRef]
6. Snyder, D.; Garcia-Romero, D.; Povey, D.; Khudanpur, S. X-vectors: Robust dnn embeddings for speaker recognition. In Proceedings of the ICASSP 2018—2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 5329–5333.
7. Gusev, A.; Volokhov, V.; Andzhukaev, T.; Novoselov, S.; Lavrentyeva, G.; Volkova, M.; Gazizullina, A.; Shulipa, A.; Gorlanov, A.; Avdeeva, A.; et al. Deep Speaker Embeddings for Far-Field Speaker Recognition on Short Utterances. *Odyssey* **2020**, *2020*, 179–186. [CrossRef]
8. Lavechin, M.; Gill, M.P.; Bousbib, R.; Bredin, H.; Garcia-Perera, L.P. End-to-end Domain-Adversarial Voice Activity Detection. *arXiv* **2019**, arXiv:1910.10655.
9. Lavrentyeva, G.; Volkova, M.; Avdeeva, A.; Novoselov, S.; Gorlanov, A.; Andzhukaev, T.; Ivanov, A.; Kozlov, A. Blind Speech Signal Quality Estimation for Speaker Verification Systems. In Proceedings of the Interspeech 2020, Shanghai, China, 25–29 October 2020; pp. 1535–1539. [CrossRef]
10. AMI Corpus. Available online: https://groups.inf.ed.ac.uk/ami/corpus/ (accessed on 30 May 2021).
11. Andrei, V.; Cucu, H.; Burileanu, C. Overlapped Speech Detection and Competing Speaker Counting—Humans Versus Deep Learning. *IEEE J. Sel. Top. Signal Process.* **2019**, *13*, 850–862. [CrossRef]
12. Chung, J.S.; Lee, B.J.; Han, I. Who said that?: Audio-visual speaker diarisation of real-world meetings. *arXiv* **2019**, arXiv:1906.10042.
13. Desplanques, B.; Thienpondt, J.; Demuynck, K. ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification. *arXiv* **2020**, arXiv:2005.07143v3.
14. Kumar, A.K.; Waldekar, S.; Saha, G.; Sahidullah, M. Domain-Dependent Speaker Diarization for the Third DIHARD Challenge. *arXiv* **2021**, arXiv:2101.09884.
15. Ferrés, M.; Bourlard, H. Speaker diarization and linking of large corpora. In Proceedings of the 2012 IEEE Spoken Language Technology Workshop (SLT), Miami, FL, USA, 2–5 December 2012; pp. 280–285.
16. Cristia, A.; Ganesh, S.; Casillas, M.; Ganapathy, S. Talker diarization in the wild: The case of child-centered daylong audio-recordings. In Proceedings of the Interspeech 2018, Hyderabad, India, 2–6 September 2018; pp. 2583–2587.
17. von Luxburg, U. A Tutorial on Spectral Clustering. *arXiv* **2007**, arXiv:0711.0189.
18. Bissig, P.; Foerster, K.; Tanner, S.; Wattenhofer, R. Distributed discussion diarisation. In Proceedings of the 2017 14th IEEE Annual Consumer Communications I & Networking Conference (CCNC), Las Vegas, NV, USA, 8–11 January 2017; pp. 1032–1037.
19. SileroTeam. Silero VAD: Pre-Trained Enterprise-Grade Voice Activity Detector (VAD), Number Detector and Language Classifier. 2021. Available online: https://github.com/snakers4/silero-vad (accessed on 29 June 2021).
20. Povey, D.; Ghoshal, A.; Boulianne, G.; Burget, L.; Glembek, O.; Goel, N.; Hannemann, M.; Motlicek, P.; Qian, Y.; Schwarz, P.; et al. The Kaldi speech recognition toolkit. In Proceedings of the IEEE 2011 Workshop on Automatic Speech Recognition and Understanding, Waikoloa, HI, USA, 11–15 December 2011.
21. Dawalatabad, N.; Ravanelli, M.; Grondin, F.; Thienpondt, J.; Desplanques, B.; Na, H. ECAPA-TDNN Embeddings for Speaker Diarization. *arXiv* **2021**, arXiv:2104.01466.
22. Landini, F.; Profant, J.; Diez, M.; Burget, L. Bayesian HMM clustering of x-vector sequences (VBx) in speaker diarization: Theory, implementation and analysis on standard tasks. *Comput. Speech Lang.* **2022**, *71*, 101254. [CrossRef]
23. Nagrani, A.; Chung, J.S.; Zisserman, A. VoxCeleb: A large-scale speaker identification dataset. In Proceedings of the Interspeech 2017, Stockholm, Sweden, 20–24 August 2017.
24. Chung, J.S.; Nagrani, A.; Zisserman, A. VoxCeleb2: Deep Speaker Recognition. In Proceedings of the Interspeech 2018, Hyderabad, India, 2–6 September 2018.
25. Nagrani, A.; Chung, J.S.; Xie, W.; Zisserman, A. Voxceleb: Large-scale Speaker Verification in the Wild. *Comput. Speech Lang.* **2020**, *60*, 101027. [CrossRef]
26. Sadjadi, S.O.; Greenberg, C.; Singer, E.; Reynolds, D.; Mason, L.; Hernandez-Cordero, J. The 2019 NIST Speaker Recognition Evaluation CTS Challenge. In Proceedings of the Odyssey 2020 The Speaker and Language Recognition Workshop, Tokyo, Japan, 1–5 November 2020, pp. 266–272. [CrossRef]
27. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv* **2015**, arXiv:1505.04597.
28. Shruti, J. A survey of loss functions for semantic segmentation. In Proceedings of the 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), Via del Mar, Chile, 27–29 October 2020.
29. Brümmer, N.; De Villiers, E. The speaker partitioning problem. In Proceedings of the Odyssey 2010, the Speaker and Language Recognition Workshop, Brno, Czech Republic, 28 June–1 July 2010; p. 34.
30. Stöter, F.R.; Chakrabarty, S.; Edler, B.; Habets, E.A. Classification vs. regression in supervised learning for single channel speaker count estimation. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 436–440.

31. Cornell, S.; Omologo, M.; Squartini, S.; Vincent, E. Detecting and counting overlapping speakers in distant speech scenarios. In Proceedings of the Interspeech 2020, Shanghai, China, 25–29 October 2020.
32. Yousefi, M.; Hansen, J. Real-time Speaker counting in a cocktail party scenario using Attention-guided Convolutional Neural Network. *arXiv* **2021**, arXiv:2111.00316.
33. Stöter, F.R.; Chakrabarty, S.; Edler, B.; Habets, E.A. CountNet: Estimating the number of concurrent speakers using supervised learning. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2018**, *27*, 268–282. [CrossRef]
34. Timofeeva, E.; Evseeva, E.; Zaluskaia, V.; Kapranova, V.; Astapov, S.; Kabarov, V. Improvement of Speaker Number Estimation by Applying an Overlapped Speech Detector. In *Speech and Computer*; Karpov, A.; Potapova, R., Eds.; Springer International Publishing: Cham, Switzerland, 2021; pp. 692–703.
35. Bredin, H.; Yin, R.; Coria, J.M.; Gelly, G.; Korshunov, P.; Lavechin, M.; Fustes, D.; Titeux, H.; Bouaziz, W.; Gill, M.P. Pyannote. audio: Neural building blocks for speaker diarization. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–9 May 2020; pp. 7124–7128.
36. Bullock, L.; Bredin, H.; Garcia-Perera, L.P. Overlap-aware diarization: Resegmentation using neural end-to-end overlapped speech detection. In Proceedings of the ICASSP 2020—2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–9 May 2020; pp. 7114–7118.
37. Kunešová, M.; Hrúz, M.; Zajíc, Z.; Radová, V. Detection of overlapping speech for the purposes of speaker diarization. In Proceedings of the International Conference on Speech and Computer, Istanbul, Turkey, 20–25 August 2019; pp. 247–257.
38. Otterson, S.; Ostendorf, M. Efficient use of overlap information in speaker diarization. In Proceedings of the 2007 IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU), Kyoto, Japan, 9–13 December 2007; pp. 683–686.
39. Boakye, K.; Trueba-Hornero, B.; Vinyals, O.; Friedland, G. Overlapped speech detection for improved speaker diarization in multiparty meetings. In Proceedings of the 2008 IEEE International Conference on Acoustics, Speech and Signal Processing, Las Vegas, NV, USA, 30 March–4 April 2008; pp. 4353–4356.
40. Alexeev, A.; Kukharev, G.; Matveev, Y.; Matveev, A. A highly efficient neural network solution for automated detection of pointer meters with different analog scales operating in different conditions. *Mathematics* **2020**, *8*, 1104. [CrossRef]
41. Park, D.S.; Chan, W.; Zhang, Y.; Chiu, C.C.; Zoph, B.; Cubuk, E.D.; Le, Q.V. SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. In Proceedings of the Interspeech 2019, Graz, Austria, 15–19 September 2019; pp. 2613–2617. [CrossRef]
42. Ravanelli, M.; Parcollet, T.; Plantinga, P.; Rouhe, A.; Cornell, S.; Lugosch, L.; Subakan, C.; Dawalatabad, N.; Heba, A.; Zhong, J.; et al. SpeechBrain: A General-Purpose Speech Toolkit. *arXiv* **2021**, arXiv:2106.04624.
43. Brock, A.; De, S.; Smith, S.L.; Simonyan, K. High-Performance Large-Scale Image Recognition Without Normalization. *arXiv* **2021**, arXiv:2102.06171.
44. Huang, Y.; Wang, Y.; Tai, Y.; Liu, X.; Shen, P.; Li, S.; Li, J.; Huang, F. CurricularFace: Adaptive Curriculum Learning Loss for Deep Face Recognition. *arXiv* **2020**, arXiv:2004.00288.
45. Gusev, A.; Vinogradova, A.; Novoselov, S.; Astapov, S. SdSVC Challenge 2021: Tips and Tricks to Boost the Short-Duration Speaker Verification System Performance. In Proceedings of the Interspeech 2021, Brno, Czech Republic, 30 August–3 September 2021. [CrossRef]
46. van Leeuwen, D.; Brümmer, N. An Introduction to Application-Independent Evaluation of Speaker Recognition Systems. In *Speaker Classification I. Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2007; Volume 4343._19. [CrossRef]
47. Haasnoot, E.; Khodabakhsh, A.; Zeinstra, C.; Spreeuwers, L.; Veldhuis, R. FEERCI: A Package for Fast Non-Parametric Confidence Intervals for Equal Error Rates in Amortized O(m log n). In Proceedings of the 2018 International Conference of the Biometrics Special Interest Group (BIOSIG), Darmstadt, Germany, 26–28 September 2018; pp. 1–5. [CrossRef]
48. Biagetti, G.; Crippa, P.; Falaschetti, L.; Orcioni, S.; Turchetti, C. Robust Speaker Identification in a Meeting with Short Audio Segments. In *Intelligent Decision Technologies 2016*; Czarnowski, I., Caballero, A.M., Howlett, R.J., Jain, L.C., Eds.; Springer International Publishing: Cham, Switzerland, 2016; pp. 465–477.