Exploring Speaker-Related Information in Spoken Language Understanding for Better Speaker Diarization

Luyao Cheng, Siqi Zheng, Zhang Qinglin, Hui Wang, Yafeng Chen, Qian Chen
Speech Lab of DAMO Academy, Alibaba Group
{shuli.cly,zsq174630,qinglin.zql,tanqing.cq}@alibaba-inc.com

Abstract

Speaker diarization(SD) is a classic task in speech processing and is crucial in multi-party scenarios such as meetings and conversations. Current mainstream speaker diarization approaches consider acoustic information only, which result in performance degradation when encountering adverse acoustic conditions. In this paper, we propose methods to extract speaker-related information from semantic content in multi-party meetings, which, as we will show, can further benefit speaker diarization. We introduce two sub-tasks, **Dialogue Detection** and **Speaker-Turn Detection**, in which we effectively extract speaker information from conversational semantics. We also propose a simple yet effective algorithm to jointly model acoustic and semantic information and obtain speaker-identified texts. Experiments on both AISHELL-4 and AliMeeting datasets show that our method achieves consistent improvements over acoustic-only speaker diarization systems.

1 Introduction

Speaker diarization(SD) is the task of answering the question “who speaks when" by partitioning audio into segments with speaker identities. In most application settings, the results of speaker diarization are perceived by readers through the assignment of speaker labels to the corresponding words or sentences transcribed from an Automatic Speech Recognition(ASR) system.

Despite the rich profusion of transcribed texts, mainstream speaker diarization systems consider only acoustic information (Park et al., 2021; Horiguchi et al., 2020; Park et al., 2020; Fujita et al., 2019; Zheng et al., 2021; Du et al., 2022a). Traditional SD systems usually consist of the following components: (1) Voice activity detection (VAD) to filter out non-speech frames. (2) Extraction of speaker embeddings from the short audio segments, using popular models such as i-vector(Dehak et al., 2011), d-vector(Zhang and Koishida, 2017) and x-vector(Snyder et al., 2018). (3) Clustering embeddings into several classes using algorithms such as agglomerative hierarchical clustering (AHC) (Day and Edelsbrunner, 1984), spectral clustering(SC) (Wang et al., 2017), and HDBSCAN (Zheng et al., 2022). Various speaker embedding model and clustering methods have been explored and proposed in (Yu et al., 2021; Dawalatabad et al., 2021; He et al., 2021; Zheng and Suo, 2022; Du et al., 2022b).

Utilizing only acoustic information has significant limitations. For example, the performance of SD system suffers from obvious degradation in adverse acoustic conditions such as noise, reverberation, and far-field recordings. In addition, we often encounter speakers with similar voice characteristics, which pose serious challenge to clustering them into expected classes. Given the abundance of transcribed texts present in meetings and conversations, it is of sufficient interest to explore the possibilities of utilizing semantic information to go beyond the limits of acoustic-only speaker diarization.

Some previous works tried to use semantic information to classify roles in two-speaker conversations, such as doctor-patient conversation and pilot-air traffic controller dialogue(Zuluaga-Gomez et al., 2021; Flemotomos and Narayanan, 2022). However, these methods are only suitable for specific two-speaker scenarios where the roles are clearly-defined, such as medical diagnosis, job interviews, and air traffic communications. In this work we focus on open multi-party meeting scenarios where the number of speakers is unknown and the relations between speakers are unspecified.

Speaker identity information has been proven to be beneficial to many downstream NLP tasks(Chi et al., 2017a; Zhang et al., 2018; Chi et al., 2017b). However, these works only consider speaker identities as given ground truth (Carletta et al., 2006; Janin et al., 2003; Zhong et al., 2021), which is

Findings of the Association for Computational Linguistics: ACL 2023, pages 14068–14077
July 9-14, 2023

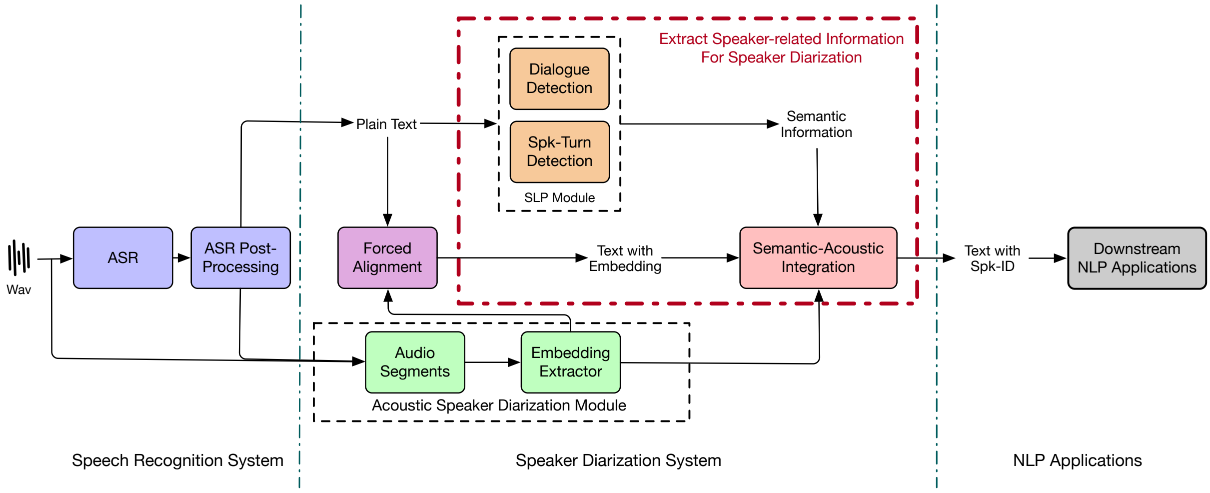

Figure 1: We propose a multi-modal speaker diarization system that utilizes the SLP module to extract speaker-related information from transcribed text. The multi-modal fusion and semantic backend modules combine both acoustic and semantic information to improve the accuracy of speaker diarization. The system's output includes text segments with corresponding speaker identification.

impractical in real world settings. Therefore, it is crucial to make valid inference of speaker identities on the transcribed conversations using a well-performed speaker diarization system.

The main contributions of this paper include:

(1) We propose two semantic tasks to extract speaker-related information from automatically transcribed texts, namely Dialogue Detection and Speaker-Turn Detection.

(2) We design a simple yet effective integration method to effectively combine semantic and acoustic information for more robust speaker diarization.

2 Proposed Methods

2.1 A Novel Multi-modal Framework

Figure 1 illustrates the proposed semantic-acoustic speaker diarization system, along with its relation with upstream ASR components and downstream NLP applications. We introduce a Spoken Language Processin(SLP) module involving two sub-tasks to extract speaker-related information from transcribed texts. The acoustic-based speaker diarization system is used to process original audio, perform segmentation and estimate speaker embeddings for each segments. To associate speaker embeddings with corresponding text phrases, a forced alignment component was introduced to our system. Finally, we propose an integration method to collectively process outputs from SLP module, acoustic SD module, and forced alignment module.

2.2 Learning Speaker Information From Texts

To extract semantic speaker-related information, we define two sub-tasks: **dialogue detection** and **speaker-turn detection**.

Dialogue detection takes a sequence of sentences as input and determines whether this is transcribed from a multi-speaker dialogue or a single-speaker speech. Dialogue-detection can be defined as a binary classification problem.

Speaker turn detection tries to predict, for each given sentence in the sequence, the probability of the occurrence of speaker change. Speaker turn detection can be defined as a sequence labeling problem, where the goal is to determine whether the given position represents a point of change in speaker role from a semantic perspective.

Both dialogue detection and speaker turn detection models are fine-tuned from a pre-trained BERT language model. Design of training samples and details of experiments are discussed in next section.

2.3 Integrating Semantic-Acoustic Information

In this section we describe how speaker-related information extracted from semantic content can assist us in improving upon acoustic-only SD system. A traditional SD system typically involves an audio segmentation module and an embedding clustering module. Poor segmentation and incorrect clustering are the most common problems in speaker diarization. Semantic information from dialogue detection helps improve clustering accuracy and speaker turn detection helps to find more precise place in text where a change of speaker occurs.

Note that dialogue detection and speaker turn detection tasks can be solved either by acoustic-only approach or semantic-only approach. Semantic-only approach is described above. Acoustic-only results can be derived directly from acoustic-based speaker clustering. The speaker clustering algorithm assigns a cluster label to each speaker-segment. Acoustic results for dialogue detection can be obtained simply by checking whether the number of different speaker labels is larger than 1. Results for speaker turn detection can be obtained by analyzing the transition patterns of speaker-segment labels or predicting change points using an acoustic-based neural networks such as Target-Speaker VAD(He et al., 2021).

Semantic-Acoustic Dialogue Detection. Let $z^{(s)}$ denote the result of binary classification output of semantic dialogue detection and $z^{(a)}$ be the counterpart of acoustic dialogue detection. We also define D_p to be the distance of the largest speaker cluster present in the dialogue to its furthest cluster, and D_q to be the standard deviation of the cosine distances among all speaker embeddings present in the selected speech. D_p measures how spread out different clusters are and D_q measures how tight embeddings in one cluster are grouped together. Then the fusion score $\hat{s}$ is estimated by:

$$\hat{s} = z^{(a)} z^{(s)} + z^{(a)}(p_s + \alpha_1 D_p) + z^{(s)}(p_s + \alpha_2 D_q), \quad (1)$$

where α_1 and α_2 are learnable and p_s is logit output from semantic dialogue detection.

For some threshold θ, the binary output of semantic-acoustic dialogue detection is represented by the indicator function:

$$\hat{z}_{dd}^{\text{fusion}} = \mathbf{1}_{\hat{s} > \theta} \quad (2)$$

Once semantic-acoustic dialogue detection obtain results for all sentence sequences that cover the entire transcribed meeting, we re-adjust the acoustic-based clustering results. By doing this we are able to incorporate semantic information to improve speaker clustering. More details can be found in Appendix B.

Semantic-Acoustic Speaker Turn Detection. Semantic-only speaker turn detection outputs a sequence of probability of the occurrence of speaker change. Let p_n be the probability at position n, and q_n represents the speaker change probability from an acoustic-only model near position n. q_n is obtained by taking the maximum probability of the closest 200 frames estimated by the Target-Speaker VAD model. Then the integrated speaker-change probability is given by

$$\tilde{p}_n = \beta_1 p_n + \beta_2 q_n \quad (3)$$

for some learnable hyperparameters β_1 and β_2.

Boundary and Outlier Correction. We use semantic information to correct boundary errors caused by errors and mismatches from the forced-alignment and ASR models. We also use semantic information to correct outliers in embedding extraction. To improve system robustness, we exclude audio segments that are too short from clustering. Outliers and left-out embeddings are assigned to the closest cluster.

	AISHELL-4 Train/Eval	Alimeeting Train/Eval
Session	191/20	209/20
#Avg. Duration(s)	1939.03/2245.94	1915.52/1924.7
#Avg. Speakers	4.8/5.8	3.27/3.0
#Avg. Speaker-Turn	343.95/220.8	649.34/552.75
Avg. Text Len.	8904.8/9990.9	13249.55/12067.05

Table 1: Details of AISHELL-4 and AliMeeting data.

3 Experiments and Results

3.1 Datasets

We conduct experiments on AISHELL-4(Fu et al., 2021) and AliMeeting(Yu et al., 2022) datasets. Both focus on multi-party meeting scenario, where all speech content are manually annotated. Table 1 listed detailed information about the datasets. We perform experiments using both ground truth (GT) text and text transcribed from ASR system.

3.2 Experimental Setups

In our experiments, the acoustic modules, including ASR, ASR Post-Processing, Embedding Extractor, and Forced Alignment models, are fixed and used consistently throughout all our experiments. In details, the ASR system we introduced was based on UniASR(Gao et al., 2020). The ASR Post-Processing contained Punctuation-Prediction(Chen et al., 2020) and Text-Smoothing which are common used in meeting scenrio. The Forced Alignment module we introduced was

TaskName	Text	Methods	AISHELL-4			Alimeeting		
			Precision	Recall	F1	Precision	Recall	F1
Dialogue Detection	GT	Acoustic-Only	74.402	84.995	79.346	93.012	92.259	92.634
		Semantic-Only	74.649	96.976	84.360	94.669	98.009	96.310
		Multi-Modal	86.308	93.402	**89.715**	96.450	97.600	**97.020**
	ASR	Acoustic-Only	80.405	96.936	87.900	96.482	98.428	97.445
		Semantic-Only	55.731	84.414	67.138	93.649	88.688	91.101
		Multi-Modal	82.461	95.826	**88.642**	96.641	98.320	**97.476**
Speaker-Turn Detection	GT	Acoustic-Only	53.962	51.272	52.583	54.329	52.997	53.655
		Semantic-Only	69.569	89.514	78.291	76.696	93.141	**84.123**
		Multi-Modal	81.652	77.240	**79.385**	76.861	92.849	84.102
	ASR	Acoustic-Only	61.657	79.162	69.322	66.105	67.696	66.891
		Semantic-Only	42.299	63.386	50.738	61.046	50.100	55.034
		Multi-Modal	68.132	73.878	**70.889**	67.593	66.960	**67.276**

Table 2: The results of two sub-tasks on AISHELL-4 and Alimeeting test set.

Text Type	Methods	AISHELL-4			Alimeeting		
		cp-wer	cp-wer-all	speaker-wer	cp-wer	cp-wer-all	speaker-wer
GT Text	Baseline - only acoustic info.	17.309	19.099	**5.974**	41.669	52.617	18.888
	Semantic-acoustic info. - A	15.540	18.798	6.558	36.360	45.772	14.700
	Semantic-acoustic info. - B	**15.225**	**18.364**	6.281	**36.145**	**45.462**	**14.500**
ASR Text	Baseline - only acoustic info.	33.905	35.590	3.647	45.678	49.778	8.404
	Semantic-acoustic info. - A	33.355	34.318	2.650	38.467	40.182	2.413
	Semantic-acoustic info. - B	**33.290**	**34.210**	**2.575**	**38.459**	**40.154**	**2.389**

Table 3: The speaker diarization results of different systems on the AISHELL-4 and Alimeeting test set

based on (McAuliffe et al., 2017). For acoustic speaker diarization system, we employed a speaker embedding extractor based on ECAPA-TDNN(Desplanques et al., 2020), while for speaker clustering, we utilized Spectral Clustering algorithm with p-percentile(Wang et al., 2017).

We fine-tune the semantic models for dialogue detection and speaker turn detection tasks based on the pre-trained BERT language model[1] using the text from AISHELL-4 and AliMeeting training sets. Training samples are sequences of sentences generated by a sliding-window method with a window length of 64 and a shift of 16 and the label for these two semantic subtasks can be generated by the speaker label from the speech content manually annotated in the datasets.

For dialogue detection task, we fine-tune for 3 epochs on train dataset with a learning rate of 5e-6 and a batch size of 64. For speaker-turn detection task, we also fine-tune for 3 epochs on train dataset with a learning rate of 1e-6 and a batch size of 64.

[1]Based on bert-base-chinese from HuggingFace

3.3 Results and Discussions

We compare our proposed methods with the classic speaker diarization system mentioned in Section 1.

Table 2 shows the results of dialogue detection and speaker-turn detection tasks from acoustic-only, semantic-only, and multi-modal models. We not only compare results using ASR-transcribed texts, but also conduct experiments using ground truth texts as inputs, in order to see the optimal improvements introduced by semantic information. The multi-modal model surpasses single-modal results on both GT and ASR text. The experiments demonstrate that semantic model can effectively supplement acoustic-only model, resulting in more precise speaker representation. It is expected that the introduction of semantic information on ASR text does not result in a significant improvement due to the a lot of errors present in the text. However, our multi-modal approach shows consistent improvement in both GT and ASR-based results, indicating the robustness of our method.

We use the **cp-WER** metric to measure the speaker diarization task. We introduce a new metric **speaker-WER** that aims to measure the word error rate caused **solely** by speaker errors. More details about the metrics can be found in Appendix A. Table 3 shows the final results of our speaker diarization system. Compared to the baseline, which only uses acoustic information, our system incorporating semantic information shows a significant improvement. The results for “Semantic-acoustic info. - A" indicate that only semantic information is used for sub-tasks Dialogue Detection and Speaker Turn Detection, while “Semantic-acoustic info. - B" indicates that both semantic and acoustic information are used in the two sub-tasks.

4 Conclusions

We propose a novel multi-modal speaker diarization system that utilizes two spoken language processing tasks, dialogue detection and speaker-turn detection, to extract speaker-related information from text. These information are then combined with acoustic information to improve the overall performance of speaker diarization. Our experiments demonstrate that incorporating semantic information can effectively address the limitations of single-modal speech.

5 Limitations

The performance of SLP tasks rely heavily on the accuracy of ASR system. Poorly-transcribed texts can lead to degradation of our multi-modal method. Since we cannot easily obtain accurate speaker-turn information from the ASR text, the training set for SLP tasks based on ASR text is also not easy to obtain. In future work, we will try more methods, like Data Arguments, to get better results on ASR text.

Overlapping speech is another challenge for the task, as a monaural ASR system can no longer capture all spoken words from all speakers. In future work, we plan to explore methods such as speech separation or multi-party ASR to handle overlapping speech.

References

Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, et al. 2006. The ami meeting corpus: A pre-announcement. In *International workshop on machine learning for multimodal interaction*, pages 28–39. Springer.

Qian Chen, Mengzhe Chen, Bo Li, and Wen Wang. 2020. Controllable time-delay transformer for real-time punctuation prediction and disfluency detection. *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8069–8073.

Ta-Chung Chi, Po-Chun Chen, Shang-Yu Su, and Yun-Nung Chen. 2017a. Speaker role contextual modeling for language understanding and dialogue policy learning. *IJCNLP 2017*, page 163.

Ta-Chung Chi, Po-Chun Chen, Shang-Yu Su, and Yun-Nung (Vivian) Chen. 2017b. Speaker role contextual modeling for language understanding and dialogue policy learning. In *International Joint Conference on Natural Language Processing*.

Nauman Dawalatabad, Mirco Ravanelli, Franccois Grondin, Jenthe Thienpondt, Brecht Desplanques, and Hwidong Na. 2021. Ecapa-tdnn embeddings for speaker diarization. In *Interspeech*.

William H. E. Day and Herbert Edelsbrunner. 1984. Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of Classification*, 1:7–24.

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

Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck. 2020. Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification. In *Interspeech*.

Zhihao Du, Shiliang Zhang, Siqi Zheng, and Zhijie Yan. 2022a. Speaker embedding-aware neural diarization: an efficient framework for overlapping speech diarization in meeting scenarios. *CoRR*, abs/2203.09767.

Zhihao Du, Shiliang Zhang, Siqi Zheng, and Zhijie Yan. 2022b. Speaker overlap-aware neural diarization for multi-party meeting analysis. *CoRR*, abs/2211.10243.

Nikolaos Flemotomos and Shrikanth S. Narayanan. 2022. Multimodal clustering with role induced constraints for speaker diarization. In *INTERSPEECH*.

Yihui Fu, Luyao Cheng, Shubo Lv, Yukai Jv, Yuxiang Kong, Zhuo Chen, Yanxin Hu, Lei Xie, Jian Wu, Hui Bu, et al. 2021. Aishell-4: An open source dataset for speech enhancement, separation, recognition and speaker diarization in conference scenario. *arXiv preprint arXiv:2104.03603*.

Yusuke Fujita, Naoyuki Kanda, Shota Horiguchi, Yawen Xue, Kenji Nagamatsu, and Shinji Watanabe. 2019. End-to-end neural speaker diarization with self-attention. *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 296–303.

Zhifu Gao, Shiliang Zhang, Ming Lei, and Ian Mcloughlin. 2020. Universal asr: Unifying streaming and non-streaming asr using a single encoder-decoder model. *ArXiv*, abs/2010.14099.

Maokui He, Desh Raj, Zili Huang, Jun Du, Zhuo Chen, and Shinji Watanabe. 2021. Target-speaker voice activity detection with improved i-vector estimation for unknown number of speaker. In *Interspeech*.

Shota Horiguchi, Yusuke Fujita, Shinji Watanabe, Yawen Xue, and Kenji Nagamatsu. 2020. End-to-end speaker diarization for an unknown number of speakers with encoder-decoder based attractors. *ArXiv*, abs/2005.09921.

Adam L. Janin, Don Baron, Jane Edwards, Daniel P. W. Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke, and Chuck Wooters. 2003. The icsi meeting corpus. *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, 1:I–I.

Michael McAuliffe, Michaela Socolof, Sarah Mihuc, Michael Wagner, and Morgan Sonderegger. 2017. Montreal forced aligner: Trainable text-speech alignment using kaldi. In *Interspeech*.

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

Tae Jin Park, Naoyuki Kanda, Dimitrios Dimitriadis, Kyu J. Han, Shinji Watanabe, and Shrikanth S. Narayanan. 2021. A review of speaker diarization: Recent advances with deep learning. *ArXiv*, abs/2101.09624.

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

Quan Wang, Carlton Downey, Li Wan, P. A. Mansfield, and Ignacio Lopez-Moreno. 2017. Speaker diarization with lstm. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5239–5243.

Fan Yu, Shiliang Zhang, Yihui Fu, Lei Xie, Siqi Zheng, Zhihao Du, Weilong Huang, Pengcheng Guo, Zhijie Yan, Bin Ma, Xin Xu, and Hui Bu. 2022. M2met: The icassp 2022 multi-channel multi-party meeting transcription challenge. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6167–6171.

Ya-Qi Yu, Siqi Zheng, Hongbin Suo, Yun Lei, and Wu-Jun Li. 2021. Cam: Context-aware masking for robust speaker verification. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2021, Toronto, ON, Canada, June 6-11, 2021*, pages 6703–6707. IEEE.

Chunlei Zhang and Kazuhito Koishida. 2017. End-to-end text-independent speaker verification with triplet loss on short utterances. In *Interspeech*.

Rui Zhang, Honglak Lee, Lazaros Polymenakos, and Dragomir Radev. 2018. Addressee and response selection in multi-party conversations with speaker interaction rnns. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Siqi Zheng, Weilong Huang, Xianliang Wang, Hongbin Suo, Jinwei Feng, and Zhijie Yan. 2021. A real-time speaker diarization system based on spatial spectrum. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2021, Toronto, ON, Canada, June 6-11, 2021*, pages 7208–7212. IEEE.

Siqi Zheng and Hongbin Suo. 2022. Reformulating speaker diarization as community detection with emphasis on topological structure. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022, Virtual and Singapore, 23-27 May 2022*, pages 8097–8101. IEEE.

Siqi Zheng, Hongbin Suo, and Qian Chen. 2022. PRISM: pre-trained indeterminate speaker representation model for speaker diarization and speaker verification. In *Interspeech 2022, 23rd Annual Conference of the International Speech Communication Association, Incheon, Korea, 18-22 September 2022*, pages 1431–1435. ISCA.

Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan Awadallah, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, and Dragomir R. Radev. 2021. Qmsum: A new benchmark for query-based multi-domain meeting summarization. In *North American Chapter of the Association for Computational Linguistics*.

Juan Zuluaga-Gomez, Seyyed Saeed Sarfjoo, Amrutha Prasad, Iuliia Nigmatulina, Petr Motlícek, Karel Ondrej, Oliver Ohneiser, and Hartmut Helmke. 2021. Bertraffic: Bert-based joint speaker role and speaker change detection for air traffic control communications.

A Metrics

The performance metrics for speaker diarization is concatenated minimum-permutation Word error rate(cpWER). The cpWER is computed as follows:

1. Concatenate each speaker's utterances both from reference and hypothesis results.

2. Compute the WER between the reference and all possible speaker permutation of hypothesis.

3. Choose the lowest WER among the results from all the speaker permutations as the final cpWER.

Specifically, the Word Error Rate(WER) is calculated by:

$$E_{wer} = \frac{N_{Ins} + N_{Subs} + N_{Del}}{N_{Total}} \times 100\% \quad (4)$$

The cpWER is affected both by the speaker diarization system and speech recognition system.

Note that the number of speakers in the system result and the reference result are not equal. If we ignore the cpWER measured by this part of the text, we will record it as $E_{cp-matched}$. If we think that this part of the text should be considered all wrong, we will record the cpWER this time for E_{cp-all}.

We know that WER calculates the minimum edit distance from the system result to the reference result. In the calculation process, the cost of changing one sequence into another sequence by using three operations of insertion, deletion and replacement is counted, while the calculation of cpWER additionally introduces the operation of modifying the speaker ID of a word to make two speaker-labeled texts become consistent.

Since the speaker diarization system cannot modify the text results of speech recognition, we calculate the speaker-WER by removing the errors caused by ASR results from cpWER. Compared to the three string operations in WER, we additionally define an operation to convert one text result with speaker ID to another by modifying the speaker ID of a word. Similar to the WER algorithm, we use dynamic programming to count the number of operations for changing the speaker ID. The speaker-WER results is calculated by:

$$E_{speakerWER} = \frac{N_{Spk-Cost}}{N_{Total}} \quad (5)$$

B Pseudocodes to Update Speaker Diarization from Semantic results

The following psuedocodes show how results from Dialogue Detection and Speaker Turn Detection are utilized to re-adjust and update speaker diarization results.

We create a sequence of speaker change occurrences, $P_{stp} = \{p_1, p_2, ..., p_N\}$, where N is the number of speaker changes, by combining the results of the Dialogue Detection and Speaker-Turn Detection tasks.

We propose a split process, as shown in Algorithm 1, to adjust the speaker IDs of certain segments to align the cluster results with the speaker change results as closely as possible by increasing the number of speakers appropriately.

Algorithm 1 The Semantic Split Process

Require: $P_{stp} = \{p_1, p_2, ..., p_N\}$ The set of the speaker turn point, $p_i \in \{0, 1\}$
$D = \{d_1, d_2, ..., d_N\}$ Each text segments divided by speaker change,
$E = \{e_1, e_2, ..., e_N\}$ Set of mean embedding belong to d_k,
τ_{split} The split threshold
Ensure:
$\hat{S} \leftarrow [1]$, $B \leftarrow [e_1]$, $N_{spk} \leftarrow 1$, $i \leftarrow 2$
while $i \leq N$ **do**
  dist $= f(e_i, B) \in R^{Size(B)}$
  if $p_i = 1$ **then**
    if dist $< \tau_{split}$ **then**
      $s_i = argmin_{s_i \in S}$ dist
      $\hat{S} \leftarrow \hat{S}$.append($s_i$)
    else
      $N_{spk} \leftarrow N_{spk} + 1$
      $s_i \leftarrow N_{spk}$
      $\hat{S} \leftarrow \hat{S}$.append($s_i$)
    end if
  else
    $s_i \leftarrow s_{i-1}$
    $\hat{S} \leftarrow \hat{S}$.append($s_i$)
  end if
  $B \leftarrow B \cup e_i$
end while

After the split process, a merge process is implemented to eliminate redundant speaker IDs. We consider both acoustic information, such as the similarity distance between speaker embeddings, and semantic information, such as the score differences between the merge of speakers i and j, computed from the Dialogue-Detection and Speaker-Turn Detection results of the utterances with the speaker IDs in two adjacent text segments. The pseudocode for the merge process is outlined in Algorithm 2.

Algorithm 2 The Semantic Merge Process

Require: E Set of mean embedding of all the speakers

S Set of speakers after split process

τ_{merge} the merge threshold

while Can Not Merge **do**

  cost$_{sim} \leftarrow cosine_distance(E)$

  cost$_{dd} \leftarrow$ score$_s^{\text{after}}$ − score$_s^{\text{before}}$

  $i_{merge}, j_{merge} = argmin_{i,j \in S}($cost$_{sim}$ + cost$_s)$

  if cost$_{\text{all}}[i_{merge}, j_{merge}] < \tau_{merge}$ **then**

    update S by merge speaker i_{merge}, j_{merge}

    update E by merge speaker i_{merge}, j_{merge}

  end if

end while

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 5

☐ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

☐ B1. Did you cite the creators of artifacts you used?
No response.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
No response.

C ☑ Did you run computational experiments?

Section 3

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 3

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 3

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 3

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Not applicable. Left blank.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.