Spoofed Audio Detection in Multi-Speaker Conversations: A Dataset and Parallel Modular Diarization-Detection Pipeline

Kelvin Pratama[1]
kelvinpratama0612@gmail.com

Wen-Chieh Fang[1]*
wcfang@gms.ndhu.edu.tw

[1]Department of Computer Science and Information Engineering,
National Dong Hwa University, Hualien, Taiwan

Abstract

Advances in speech synthesis make it possible to manipulate audio by altering only specific speakers' turns. Spoofing research, however, has largely targeted single-speaker audio, leaving the more realistic multi-speaker case—identifying which speaker was altered and when—underexplored, with prior work restricted to fully spoofed audio or treating diarization and detection as disconnected stages. This study introduces SpoofedConversation, a dataset of bona fide, fully spoofed, and partially spoofed multi-speaker conversations, and a parallel modular pipeline combining speaker diarization with segment-level (20 ms) spoof detection to localize and attribute spoofed regions to corresponding speaker—the first pipeline evaluated end-to-end on this task. An adapted diarization error rate (ADER) is also proposed, scoring false positive, false negative, and speaker confusion per class, exposing biases that standard DER and EER conceal. The best internal configuration lowers ADER by 15—23 percentage points over a weaker one, while revealing a systematic bias toward the spoof class—establishing the feasibility of joint spoof localization and attribution and ADER's diagnostic value.

Keywords: Spoofed Audio Detection, Multi-speaker Conversation, Partially Spoofed Audio, Speaker Diarization

1 Introduction

Audio recordings have long been trusted as faithful representations of real acoustic events, underpinning applications in communication, multimedia production, and forensics (Nasr et al., 2025).

Recent advances in generative models have enabled highly realistic synthetic speech, commonly called deepfake or spoof audio (Li et al., 2025). A more concerning variant, partially spoofed audio (Zhang et al., 2021b), alters only targeted segments of a recording while leaving the rest intact, making it especially hard to detect since most of the original acoustic characteristics and contextual coherence are preserved.

Most existing research addresses single-speaker scenarios, yet real-world interactions are typically conversational and multi-speaker, where spoofed audio can be deployed more covertly by altering only one speaker's turn while the rest of the dialogue remains authentic. This is not merely theoretical: real-world incidents such as (Sólymos, 2023) show spoofed speech already embedded within natural conversational dynamics, including interviews and debates. Despite this practical relevance, detection methods for conversational audio deepfakes remain underexplored (see Sec. 2).

The main contributions of this study are to establish the feasibility of the task and a rigorous joint evaluation methodology:

1. A novel dataset, SpoofedConversation, providing multi-speaker conversational audio with embedded spoofed speech segments.

2. A parallel modular pipeline for detecting and localizing spoofed speech segments in multi-speaker conversational audio.

3. An adapted diarization error rate (ADER) metric that jointly scores spoof detection and speaker diarization errors (Sec. 4.4.1).

* Corresponding author

Proceedings of the 38th Conference on Computational Linguistics and Speech Processing (ROCLING 2026), pages 67–76
November 6-7, 2026 ©2026 Association for Computational Linguistics

The remainder of the paper proceeds as follows: Sec. 2 reviews related work; Sec. 3 details the dataset construction and pipeline architecture; Sec. 4 describes the training configuration, baselines, and evaluation metrics; Sec. 5 presents the results; and Sec. 6, 7, and 8 conclude the study and discuss its limitations and future directions.

2 Related Work

Prior work on spoofed audio detection has largely targeted fully spoofed utterances (Tak et al., 2021; Ma et al., 2021; Chen et al., 2025). In practice, however, attackers often manipulate only specific segments of an utterance, a scenario known as partially spoofed audio, which poses a substantially harder detection problem.

Zhang et al. (2021b) showed via cross-database evaluation that models trained on fully spoofed (FS) data generalize poorly to partially spoofed (PS) data, especially for short manipulated segments, while PS-trained models generalize well to both, suggesting that FS training overfits to whole-utterance spoofing.

Several studies and challenges (Luong et al., 2025; Yi et al., 2023a; Zhang et al., 2021a, 2023a; Xie et al., 2023; Yi et al., 2024, 2023b) have proposed countermeasures for PS detection, typically alongside dedicated datasets. Zhang et al. (2023a), for example, built the PartialSpoof dataset by inserting spoofed (fake) segments into bona fide (genuine) recordings and vice versa, and paired it with a self-supervised learning (SSL)-based countermeasure that models multiple temporal resolutions, achieving low error rates at both the utterance and segment level.

Most existing work, however, targets single-speaker audio. Wijethunga et al. (2020) addressed the more complex multi-speaker setting, affected by overlapping speech, diarization errors, and background noise, via a three-stage pipeline of denoising, diarization, and utterance-level spoof detection, each a separately trained deep neural network. Because the stages were evaluated independently (task-specific) rather than as an integrated system, it remains unclear whether the pipeline is effective for realistic multi-speaker conversations.

More recently, Ahmed et al. (2026) introduced a dataset of real and fully spoofed two-speaker conversations, with synthetic turns generated entirely via TTS and separated by a short 0.5-second pause to preserve natural dialogue rhythm. Baseline experiments with LFCC-LCNN, RawNet2, and Wav2Vec 2.0 showed that conventional convolutional models struggle in multi-speaker settings, while end-to-end and self-supervised models are more robust.

Xu et al. (2026) built a dataset via a multi-agent process to keep partially spoofed conversational segments linguistically coherent, and proposed an audio-language framework that reformulates detection as structured generation, jointly localizing spoofed intervals and explaining how they alter meaning, aided by a spoof-focused encoder and in-context learning.

Overall, detection methods perform well on fully spoofed, single-speaker audio but degrade on partially spoofed recordings, and work on multi-speaker conversational settings remains limited and largely restricted to fully spoofed cases, despite real-world attacks being more likely to involve selective, partial manipulation. This gap motivates the present study: a pipeline for localizing spoofed segments and attributing them to the corresponding speaker in conversational audio.

3 Methodology

Most existing datasets are limited to single-speaker scenarios or, in the case of recent multi-speaker conversational datasets, to fully synthetic conversations—settings that do not sufficiently represent realistic attacks, where an adversary may manipulate only specific segments of a target speaker's speech. To address this gap, this study introduces SpoofedConversation, a dataset comprising bona fide, fully spoofed, and partially spoofed conversations. Unlike utterance-level detection approaches (Ahmed et al., 2026), the proposed pipeline supports segment-level detection at 20 ms resolution, with speaker diarization incorporated to localize speech regions and attribute them to their corresponding speakers.

3.1 SpoofedConversation Dataset

SpoofedConversation is derived from Llama-PartialSpoof (Luong et al., 2025), an English single-speaker corpus for partially spoofed audio detection, where spoofed content is generated by rewriting genuine recordings with Llama-3-8B-Instruct (Grattafiori et al., 2024) and synthesizing them with six TTS/VC (voice conversion) systems. Unlike PartialSpoof (Zhang et al., 2023a), Llama-PartialSpoof maintains semantic consistency through content-driven manipulation, making it more suitable for conversational dataset creation.

Conversational sessions are constructed by concatenating single-speaker clips to simulate multi-turn interactions, following the approach of speaker diarization datasets (Chen et al., 2020; Cosentino et al., 2020), as illustrated in Figure 1. Each session is constructed as follows:

1. **Sample session metadata.** Three parameters are randomly sampled: (i) **session duration**, from a truncated lognormal distribution capped at 30 seconds due to model constraints; (ii) **bona fide ratio**, from a uniform distribution over $[0, 1]$ to reflect diverse attack conditions; and (iii) **speaker identities**, selected according to the specified number of speakers per session.

2. **Generate speaker turn sequence**. Speakers are randomly selected and turn durations sampled until the target session length is reached. Each turn is labeled Bona Fide (BF), Fully Spoofed (FS), or Partially Spoofed (PS), with adjacent turns prohibited from sharing both speaker and segment type.

3. **Construct each speaker turn**. Utterances matching the turn's speaker identity and segment type are drawn from the data pool and concatenated to the target duration, removing each utterance from the pool to prevent reuse within the session.

4. **Insert inter-turn pauses**. Pauses are sampled from a truncated lognormal distribution (mean: 300 ms, standard deviation: 200 ms, median $\approx$ 250 ms, range: 50-1500 ms) following the conversational pause characteristics reported in (Reece et al., 2023), rather than a fixed pause duration used in (Ahmed et al., 2026).

5. **Generate annotations.** Diarization and spoof labels are assigned per segment; silence in BF/FS turns is labeled accordingly, while silence in PS turns defaults to bona fide, as it carries no discriminative acoustic cues.

3.2 Parallel Modular Pipeline

3.2.1 Architecture Overview

The proposed pipeline detects and localizes spoofed speech segments in multi-speaker conversations by integrating three components: a **Speaker Diarization Module** that identifies speaker turns, a **Spoof Detection Module** that assesses each segment for authenticity, and an **Output Alignment** algorithm that maps frame-level spoof labels to their corresponding speaker. The overall architecture is illustrated in Figure 2.

3.2.2 Speaker Diarization

Speaker diarization addresses the *"who spoke when"* problem by segmenting audio into speaker-homogeneous regions and attributing each to a speaker identity. Based on their architectural design (NVIDIA, 2026), speaker diarization models can be categorized into end-to-end approaches, such as (Park et al., 2025), and pipeline-based approaches, such as Pyannote (Pyannote).

3.2.3 Segment-level Spoof Detection

Segment-level spoof detection aims to identify and localize spoofed regions within an audio recording by classifying each speech segment as bona fide or spoofed. Unlike utterance-level detection, it requires frame-level analysis to capture subtle acoustic inconsistencies and accurately determine the boundaries of manipulated regions.

3.2.4 Output Alignment Algorithm

The output alignment is implemented using a simple mapping algorithm that maps spoof detection outputs with their corresponding

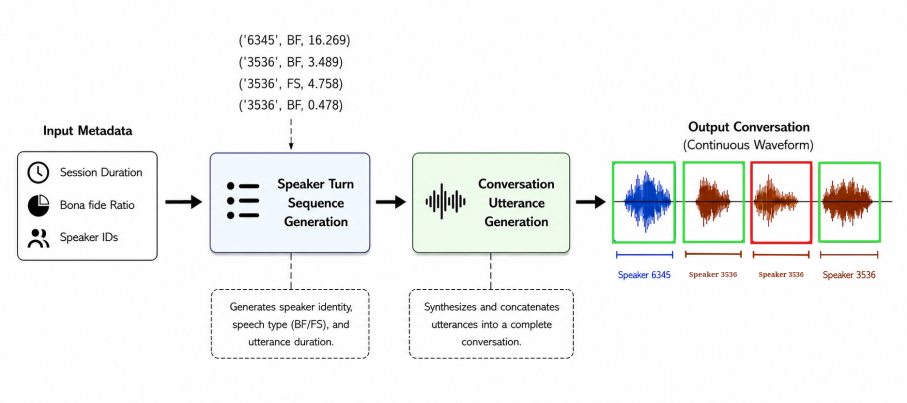

Figure 1: SpoofedConversation dataset construction pipeline

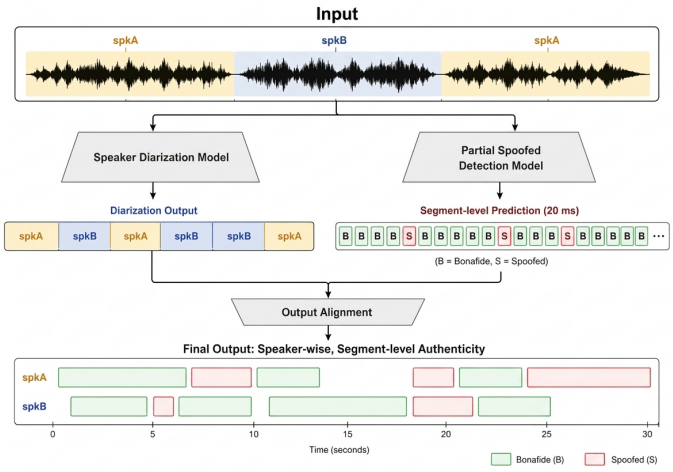

Figure 2: Parallel Modular Pipeline architecture

speech regions. Since each spoof detection label represents the authenticity of a 20 ms segment, the labels are mapped according to the temporal boundaries produced by the speaker diarization module. For example, if the diarization module detects a speech region from 0 s to 1 s, the spoof labels corresponding to that time interval are assigned to the detected speaker segment. The pseudocode of the alignment procedure is presented in Algorithm 1.

Algorithm 1 Output Alignment

Require: Diarization annotations $\mathcal{A}$, spoof segment labels $\mathcal{L}$, frame resolution $f_s = 20$ ms
Ensure: Aligned output segments $\mathcal{S}$
  Initialize $\mathcal{S} \leftarrow \emptyset$
  for all $a \in \mathcal{A}$ **do**
    $t_s \leftarrow a.start$
    $t_e \leftarrow a.end$
    $i_s \leftarrow \lfloor t_s / f_s \rfloor$
    $i_e \leftarrow \min(\lfloor t_e / f_s \rfloor, |\mathcal{L}| - 1)$
    $l_{\text{cur}} \leftarrow \mathcal{L}[i_s]$
    $t_{\text{seg}} \leftarrow t_s$
    for $i \leftarrow i_s$ to i_e **do**
      if $\mathcal{L}[i] \neq l_{\text{cur}}$ **then**
        $t_{\text{end}} \leftarrow \min(t_s + (i - i_s + 1) f_s,\ t_e)$
        Append $(t_{\text{seg}},\ t_{\text{end}},\ l_{\text{cur}},\ a.spk)$ to $\mathcal{S}$
        $l_{\text{cur}} \leftarrow \mathcal{L}[i]$
        $t_{\text{seg}} \leftarrow t_{\text{end}}$
      end if
    end for
    Append $(t_{\text{seg}},\ t_e,\ l_{\text{cur}},\ a.spk)$ to $\mathcal{S}$
  end for
  return $\mathcal{S}$

4 Experiments

4.1 Datasets

LlamaPartialSpoof (LPS) is split into speaker- and model-disjoint sets: training uses 20 speakers and four TTS systems, while evaluation uses 20 unseen speakers and two unseen TTS systems. All utterances are filtered to have shorter than 30 s duration (Table 1).

SpoofedConversation (SC) generation is automated via probabilistic sampling over

	Subset	# of Samples	Duration (h)	Max # of TTS/VC	Audio length (s)		
					Min	Mean	Max
Bona fide	Train	2,353	3.83	-	0.51	5.86	20.00
	Eval.	3,341	4.93	-	0.50	5.31	29.20
Full Spoof	Train	9,128	16.98	4	0.55	6.70	25.36
	Eval.	6,485	10.74	2	0.57	5.96	28.71
Partial Spoof	Train	8,883	15.99	4	0.60	6.48	24.94
	Eval.	6,186	9.82	2	0.59	5.72	29.35

Table 1: Statistics of LPS in speaker- and model-disjoint sets, including sample count, total duration in hours, number of TTS/VC models, and audio length (min, mean, max).

speaker count, utterance count, split, and overlap. Each SC split draws utterances only from the corresponding LPS split: SC-train from LPS-train, and SC-eval from LPS-eval. SC therefore inherits speaker- and model-disjoint of LPS. In this study, only non-overlapping sessions with one to four speakers are generated (Table 2).

4.2 Pipeline Modules

PartialSpoof (PS) (Zhang et al., 2023a) provides pretrained models at multiple temporal resolutions. This study uses the single-resolution (20 ms) and multi-resolution variants, plus the temporal deepfake location (TDL) model (Xie et al., 2023), reported to have superior performance. For speaker diarization, this study incorporates four state-of-the-art pipelines: two publicly available Pyannote pipelines, speaker-diarization-3.1 (`SD-3.1`) and speaker-diarization-community-1 (`SD-Comm-1`); the commercial Pyannote speaker-diarization-precision-2 (`SD-Prec-2`) pipeline (Pyannote; Plaquet and Bredin, 2023; Bredin, 2023); and the NVIDIA NeMo Sortformer Diarizer (Park et al., 2025).

4.3 Experimental Setup

Each component is trained/finetuned independently: spoof detectors on LPS-train, diarizers on SC-train. Spoof detection follows (Zhang et al., 2023a): P2SGrad MSE loss (Zhang et al., 2019), Adam ($\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 10^{-8}$), learning rate 1×10^{-5} halved every 10 epochs. All spoof detectors, including PS and two TDL variants (one trained with P2SGrad loss, the other with Embedding Similarity loss (Xie et al., 2023)), are finetuned for 100 epochs.

For diarization, only the segmentation model is finetuned in publicly available Pyannote pipelines (PyTorch Lightning Trainer, gradient clipping 0.5), with other components kept pretrained due to the time constraint; Sortformer, which has no separable stages, is finetuned end-to-end under the same setup. All diarizers are finetuned for 100 epochs.

The two components run as separate Docker microservices (incompatible PyTorch versions), with an output-alignment function, on a single NVIDIA RTX 4090.

4.4 Evaluation Metrics

Diarizers are evaluated on SC-eval using Diarization Error Rate (DER) (Bredin, 2017). Spoof detectors are evaluated on LPS-eval and SC-eval using Equal Error Rate (EER) (Zhang et al., 2023b), Precision, Recall, and F_1-score. The pipeline is then evaluated on SC-eval using Adapted DER (ADER).

4.4.1 Adapted DER (ADER)

To jointly evaluate spoof detection and speaker attribution, ADER is computed per class:

$$\mathrm{ADER}_{c,j} = \frac{\mathrm{FP}_{c,j} + \mathrm{FN}_{c,j} + \mathrm{Conf}_{c,j}}{T_{c,j}} \tag{1}$$

$$\mathrm{ADER}_{c} = \frac{\sum_{j \in \mathcal{D}} T_{c,j} \cdot \mathrm{ADER}_{c,j}}{\sum_{j \in \mathcal{D}} T_{c,j}} \tag{2}$$

where $c \in \{\text{bona fide}, \text{spoof}\}$, $T_{c,j}$ is the reference duration of class c in trial j, and FP/FN/Conf denote the durations of false positive, false negative, and speaker confusion errors. $\mathcal{D}$ denotes the set of all utterances. For bona fide, FP occurs when non-speech is misclassified as speech or spoof is misclassified as bona fide; FN occurs when speech is missed or bona fide is misclassified as spoof. The definitions for the spoof class are analogous. Confusion is speech attributed to the wrong speaker.

Duration-weighted averaging (Equation 2) is used rather than macro-averaging because

	# of speakers	# of Samples	Duration (h)	Audio length (s)		
				Min	Mean	Max
Train	1	34	0.21	14.19	22.60	30.00
	2	159	1.01	12.99	22.86	29.95
	3	82	0.52	10.35	22.89	29.47
	4	22	0.15	14.89	24.86	29.98
Eval.	1	47	0.28	11.40	21.43	30.00
	2	239	1.53	10.54	23.12	29.96
	3	161	1.04	10.59	23.17	29.98
	4	49	0.33	15.94	24.23	29.73

Table 2: Statistics of SC for each speaker-count configuration, including number of samples, total duration in hours, and audio length measures (min, mean, and max).

$T_{c,j}$ varies widely across trials, so per-trial rates are unstable when a class occupies little reference duration. It also matches how standard DER is computed, by accumulating errors and reference durations before dividing.

Numerical example. Consider a 10-second two-speaker utterance with error information given in Table 3.

Interval	Reference	Hypothesis	Error type
0–2	A, bona fide	A, bona fide	—
2–4	B, bona fide	**A**, bona fide	Speaker conf.
4–6	A, spoof	A, **bona fide**	Spoof→bona fide
6–8	B, bona fide	B, **spoof**	Bona fide→spoof
8–10	A, bona fide	A, bona fide	—

Table 3: Reference vs. hypothesis for the numerical example. Errors are in bold.

Reference durations are $T_{\text{bona}} = 8$s and $T_{\text{spoof}} = 2$s. For the bona fide class, the [2-4s] interval contributes $\text{Conf}_{\text{bona}} = 2$s (correct class, wrong speaker); the [4-6s] interval, where spoof is misclassified as bona fide, contributes $\text{FP}_{\text{bona}} = 2$s; and the [6-8s] interval, where bona fide is misclassified as spoof, contributes $\text{FN}_{\text{bona}} = 2$s. Hence

$$\text{ADER}_{\text{bona}} = \frac{2+2+2}{8} = 0.75.$$

For the spoof class, $T_{\text{spoof}} = 2$s covers only the [4-6s] interval. The same error at [6-8s] now counts as $\text{FP}_{\text{spoof}} = 2$s (bona fide misclassified as spoof), and the missed spoof at [4-6s] counts as $\text{FN}_{\text{spoof}} = 2$s, with no speaker confusion within spoof segments ($\text{Conf}_{\text{spoof}} = 0$):

$$\text{ADER}_{\text{spoof}} = \frac{2+2+0}{2} = 2.0.$$

This illustrates three properties distinguishing ADER from standard, class-agnostic DER:

1. The same error pair (spoof↔bona fide swaps at [4-6s] and [6-8s]) enters both $\text{ADER}_{\text{bona}}$ and $\text{ADER}_{\text{spoof}}$ making $\text{FP}_{\text{bona}}/\text{FN}_{\text{spoof}}$ complementary (likewise $\text{FN}_{\text{bona}}/\text{FP}_{\text{spoof}}$).

2. Since T_{spoof} is small relative to its errors, $\text{ADER}_{\text{spoof}}$ can exceed 1.0—rare for standard DER, where errors and reference speech share one undifferentiated pool.

3. The complementary attribute of ADER reveals detection bias toward a particular class, as can be seen in Sec. 5.4.

A standard (class-agnostic) DER on the same hypothesis, penalizing only speaker-attribution errors, yields DER = 2/10 = 0.20: a seemingly strong result concealing complete failure to detect the spoofed segment—precisely the gap ADER exposes.

4.5 Baseline Comparison

No prior system produces output directly compatible with segment-level, speaker-attributed evaluation. For spoof detection, random guessing (uniform distribution) and always-spoof (worst-case reference) are used. No comparable baseline exists for the diarization setting.

5 Results and Discussions

5.1 Baseline Performance

Table 4 shows class imbalance in both datasets, but in opposite directions at the frame level: LPS skews toward spoof frames (63%), while SC skews toward bona fide frames (62%). Both skew heavily toward spoof at the utterance level (SC more so: 87% vs. 79%), which inflates utterance-level precision under a trivial Always-spoof baseline and makes utterance-level comparison less discriminative.

Trained/finetuned models must therefore clearly exceed the 20 ms frame-level baseline.

Baseline	Metric	Random		Always-spoof	
		20ms	Utt	20ms	Utt
LPS	EER	49.99	49.56	-	-
	Precision	63.30	78.78	63.29	79.13
	Recall	50.01	49.63	100.00	100.00
	F_1	55.87	60.90	77.52	88.35
SC	EER	49.98	50.20	-	-
	Precision	38.44	87.15	38.43	86.80
	Recall	50.02	50.00	100.00	100.00
	F_1	43.47	63.54	55.52	92.93

Table 4: LPS and SC baseline results (%).

5.2 Segment-level Spoof Detection

Table 5 shows that all models struggle to detect spoofed segments on both datasets, particularly SC; precision/recall/F_1-score are omitted as they can mislead under class imbalance. `TDL-P2SGradLoss` generalizes best across both datasets, followed by `PS multireso` at the frame level, with `TDL-P2SGradLoss` also outperforming the finetuned PS variants overall. The poor performance on SC was hypothesized to stem from silence gaps, but restricting EER computation to speech-only regions yielded only a negligible improvement (≈0.4%), indicating the degradation originates from the model itself rather than silence gaps. Likely causes include (1) limited TTS coverage in the training set, (2) stronger TTS models in the evaluation set, as reflected in Table 6, and (3) context shift at boundaries, where the model encounters shifts in acoustic identity—speaker, channel, or background noise characteristics—across concatenated utterances.

Comparing the two TDL variants, `TDL-P2SGradLoss` consistently outperforms `TDL-EmbedLoss` across all evaluation in Table 5. This consistent margin indicates that P2SGrad loss is the more effective training objective for the TDL architecture.

To quantify the resulting variability, `PS multireso` and `TDL-P2SGradLoss` were retrained with five random seeds under identical configurations; results are shown in Table 7. Variability is highest for `PS multireso` on utterance-level SC, where one seed yielded chance-level EER (50%) and the remaining four ranged from 36.38% to 42.30%. As no development set was used for checkpoint selection, each run is evaluated at a fixed epoch rather than at its best state, which plausibly accounts for the spread.

Model	20 ms frame-level			Utterance-level		
	LPS	SC	SC†	LPS	SC	SC†
PS singlereso	42.14	50.28	50.37	25.35	50.35	50.35
PS multireso	**34.31**	47.38	47.15	22.36	50.00	50.00
TDL-P2SGradLoss	39.02	**44.71**	**44.33**	**14.30**	**31.81**	**31.81**
TDL-EmbedLoss	49.37	47.21	47.10	26.46	37.83	37.83

Table 5: EER (%) of trained/finetuned models on LPS-eval and SC-eval at 20 ms frame-level and utterance-level. †SC restricted to speech-only regions. Bold = best.

Model	CosyVoice	ElevenLabs
PS multireso	**39.22**	49.67
TDL-P2SGradLoss	45.51	**49.36**

Table 6: EER (%) at 20 ms frame-level of the two best models on each LPS-eval TTS subset.

Model	20 ms frame-level		Utterance-level	
	LPS	SC	LPS	SC
`multireso`	**36.14±1.57**	46.33±0.71	21.76±0.76	41.74±5.10
`tdlp2s`	36.08±3.01	**44.14±1.00**	**15.21±2.09**	**34.94±7.40**

Table 7: Mean ± SD of EER (%) of spoof detection models over five random seeds. Bold = Best

5.3 Speaker Diarization

Table 8 shows `Sortformer`[1] achieves the best DER (5.81±0.14%), outperforming all Pyannote[2] variants by a large margin, particularly in speaker confusion. However, this comparison is not fully controlled: due to time constraints, only the segmentation component of the Pyannote pipelines was finetuned, which reduces missed detection and false alarm but does not address speaker confusion. Finetuning speaker embedding and clustering components would likely narrow this gap, so the reported difference reflects finetuning depth as well as architecture.

`SD-3.1` and `SD-Comm-1` exhibit the same false alarm and missed detection errors, as both use the same segmentation module. This also indicates that, on SC-eval, the speaker embedding and clustering modules of `SD-3.1` perform slightly better than those of `SD-Comm-1`.

[1] https://huggingface.co/nvidia/diar_sortformer_4spk-v1

[2] https://huggingface.co/pyannote

Model	FA	MD	Conf	DER
SD-3.1	2.62±0.29	1.82±0.13	8.96±0.49	13.40±0.60
SD-Comm-1	2.62±0.29	1.82±0.13	9.71±0.37	14.15±0.39
SD-Prec-2†	0.36	11.74	6.48	18.59
Sortformer	**1.31±0.03**	**1.37±0.04**	**3.13±0.13**	**5.81±0.14**

Table 8: DER and its components (%) on SC-eval, mean±SD over five random seeds. † single seed. Bold = Best.

5.4 Parallel Modular Pipeline

Based on Table 5 and Table 8, `PS multireso` and `TDL-P2SGradLoss` are selected as spoof detection modules, paired with `Sortformer` as the diarization module of the pipeline. Figure 3 compares the resulting two internal configurations, under a fixed diarization module, showing that substituting `PS multireso` with `TDL-P2SGradLoss` yields absolute improvements of 15% in $\text{ADER}_{\text{bona}}$ and 23% in $\text{ADER}_{\text{spoof}}$. Considering the `Sortformer`'s performance on SC-eval in Table 8, these high errors are largely attributed to the spoof detection module, which is shown in Table 10.

	FP_{bona}	FN_{bona}	$\text{Conf}_{\text{bona}}$	$\text{ADER}_{\text{bona}}$
multireso	**19.79±1.49**	56.95±3.13	**0.72±0.22**	77.46±1.74
tdlp2s	21.84±1.69	**40.00±2.48**	1.04±0.21	**62.89±2.21**
	FP_{spoof}	FN_{spoof}	$\text{Conf}_{\text{spoof}}$	$\text{ADER}_{\text{spoof}}$
multireso	84.97±4.70	**30.28±2.20**	**2.82±0.43**	118.07±3.11
tdlp2s	**59.56±3.93**	32.90±2.37	2.99±0.26	**95.45±3.12**

Table 9: ADER and its error components (%) for bona fide (top) and spoof (bottom) classes on SC-eval, mean±SD over five random seeds.

	FP_{bona}	FN_{bona}	$\text{Conf}_{\text{bona}}$	$\text{ADER}_{\text{bona}}$
multireso	**19.70±1.48**	56.78±3.15	0	76.46±1.95
tdlp2s	21.47±1.63	**39.79±2.55**	0	**61.26±2.16**
	FP_{spoof}	FN_{spoof}	$\text{Conf}_{\text{spoof}}$	$\text{ADER}_{\text{spoof}}$
multireso	83.81±4.65	**29.07±2.18**	0	112.88±2.89
tdlp2s	**58.72±3.76**	31.69±2.41	0	**90.41±3.19**

Table 10: ADER and its error components (%) under an oracle diarization module setting.

The complementary relationship between FP and FN described in Sec. 4.4.1 can be observed here: the high $\text{ADER}_{\text{bona}}$ is dominated by FN_{bona}, while the high $\text{ADER}_{\text{spoof}}$ is dominated by FP_{spoof}. This reveals a bias in the spoof detection module toward the spoof class, illustrating ADER's ability to expose such model-specific biases.

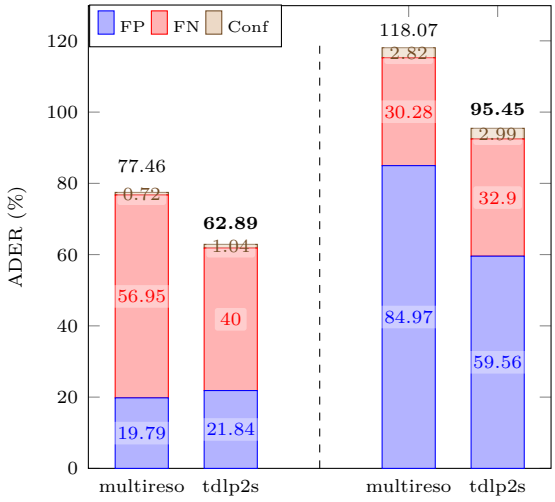

Figure 3: ADER error decomposition for bona fide (left) and spoof (right) classes on SC-eval.

6 Conclusion

This work addresses spoofed speech (audio deepfake) detection in multi-speaker conversational audio. At the time of writing, this is the first work to jointly localize and attribute partially spoofed segments to corresponding speakers, with the aligned pipeline output evaluated end-to-end. This is distinct from prior conversational spoof detection work in three ways: (Wijethunga et al., 2020) evaluates diarization and detection independently rather than as a combined system, and targets only fully spoofed multi-speaker audio; (Ahmed et al., 2026) likewise targets only fully spoofed multi-speaker audio; and (Xu et al., 2026) localizes and semantically explains spoofed intervals but does not attribute them to a specific speaker via diarization.

`TDL-P2SGradLoss` is the only model that degrades least across both datasets and across all random seeds, though its performance remains modest overall, and it outperforms Embedding loss, indicating P2SGrad is the more effective training objective for the TDL architecture. Among the diarization systems evaluated, `Sortformer` achieves the lowest DER (5.81±0.14%); however, since the Pyannote pipelines received only partial finetuning, this gap cannot be solely attributed to Sortformer's end-to-end architecture, and a controlled comparison with fully finetuned Pyannote pipelines is needed to isolate architecture from finetuning depth. The parallel modular

pipeline (`TDL-P2SGradLoss` + `Sortformer`) improves $\text{ADER}_{\text{bona}}$ by 15% and $\text{ADER}_{\text{spoof}}$ by 23% over (`PS-multireso` + `Sortformer`), and the complementary attribute of ADER further reveals a bias of the detection model toward a particular class, as shown in Figure 3.

7 Limitations

Nevertheless, several limitations constrain the current study. First, both LPS and SC lack spoofing diversity, and being read-speech English, SC also lacks natural conversational sound and structure, as utterances are concatenated without regard to contextual coherence. SC is also relatively very small, and finetuning the diarization modules for 100 epochs may increase overfitting risk. The evaluation set is further limited to only two unseen TTS systems, restricting the generalizability of findings. Finally, the proposed pipeline is currently limited to non-overlapping speech, and dependency conflicts between the spoof detection and speaker diarization modules require deployment as separate microservices, increasing implementation complexity.

8 Future Works

1. Extend SC with a broader range of state-of-the-art TTS/VC models to increase spoofing diversity.

2. Incorporate more naturalistic conversational audio, e.g., via the multi-agent writer/checker approach of (Xu et al., 2026), to improve linguistic coherence across concatenated utterances.

3. Add overlapping speech segments to SC, common in real-world conversation but currently unaddressed.

4. Explore joint training, representation sharing, or end-to-end loss optimization for a unified framework.

References

Alabi Ahmed, Vandana Janeja, and Sanjay Purushotham. 2026. Multi-speaker conversational audio deepfake: Taxonomy, dataset and pilot study.

Hervé Bredin. 2017. pyannote.metrics: A toolkit for reproducible evaluation, diagnostic, and error analysis of speaker diarization systems. In *Interspeech*.

Hervé Bredin. 2023. pyannote.audio 2.1 speaker diarization pipeline: principle, benchmark, and recipe. In *Proc. INTERSPEECH 2023*.

Dong Chen, Fan Huang, Zhengxuan Song, Wei Zhu, Yin Yang, and Kun Zeng. 2025. Wavespect: A hybrid approach to synthetic audio detection via waveform and spectrogram analysis. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Zhuo Chen, Takuya Yoshioka, Liang Lu, Tianyan Zhou, Zhong Meng, Yi Luo, Jian Wu, Xiong Xiao, and Jinyu Li. 2020. Continuous speech separation: Dataset and analysis. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7284–7288.

Joris Cosentino, Manuel Pariente, Samuele Cornell, Antoine Deleforge, and Emmanuel Vincent. 2020. Librimix: An open-source dataset for generalizable speech separation.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Menglu Li, Yasaman Ahmadiadli, and Xiao-Ping Zhang. 2025. A survey on speech deepfake detection. *ACM Comput. Surv.*, 57(7).

Hieu-Thi Luong, Haoyang Li, Lin Zhang, Kong Aik Lee, and Eng Siong Chng. 2025. Llamapartialspoof: An llm-driven fake speech dataset simulating disinformation generation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Youxuan Ma, Zongze Ren, and Shugong Xu. 2021. Rw-resnet: A novel speech anti-spoofing model using raw waveform.

Marwa A. Nasr, Walid El-Shafai, Nariman Abdel-Salam, El-Sayed M. El-Rabaie, Adel S. El-Fishawy, and Fathi E. Abd El-Samie. 2025. A comprehensive survey of audio forgery detection: challenges and novel trends. *Journal of Electrical Systems and Information Technology*, 12(1):30.

NVIDIA. 2026. Speaker diarization. Accessed: 2026-05-09.

Taejin Park, Ivan Medennikov, Kunal Dhawan, Weiqing Wang, He Huang, Nithin Rao Koluguri, Krishna C. Puvvada, Jagadeesh Balam, and

Boris Ginsburg. 2025. Sortformer: A novel approach for permutation-resolved speaker supervision in speech-to-text systems.

Alexis Plaquet and Hervé Bredin. 2023. Powerset multi-class cross entropy loss for neural speaker diarization. In *Proc. INTERSPEECH 2023*.

Pyannote. pyannote speaker diarization toolkit. https://github.com/pyannote/pyannote-audio. Accessed: 2026-05-09.

Andrew Reece, Gus Cooney, Peter Bull, Christine Chung, Bryn Dawson, Casey Fitzpatrick, Tamara Glazer, Dean Knox, Alex Liebscher, and Sebastian Marin. 2023. The candor corpus: Insights from a large multimodal dataset of naturalistic conversation. *Science Advances*, 9(13):eadf3197.

Karin Kőváry Sólymos. 2023. Slovakia: Deepfake audio of denník n journalist offers worrying example of ai abuse. Accessed: 2026-02-11.

Hemlata Tak, Jose Patino, Massimiliano Todisco, Andreas Nautsch, Nicholas Evans, and Anthony Larcher. 2021. End-to-end anti-spoofing with rawnet2.

R.L.M.A.P.C. Wijethunga, D.M.K. Matheesha, Abdullah Al Noman, K.H.V.T.A. De Silva, Muditha Tissera, and Lakmal Rupasinghe. 2020. Deepfake audio detection: A deep learning based solution for group conversations. In *2020 2nd International Conference on Advancements in Computing (ICAC)*, volume 1, pages 192–197.

Yuankun Xie, Haonan Cheng, Yutian Wang, and Long Ye. 2023. An efficient temporary deepfake location approach based embeddings for partially spoofed audio detection.

Xuenan Xu, Yiming Ren, Liwei Liu, Wen Wu, Baoxiang Li, Chaochao Lu, Shuai Wang, and Chao Zhang. 2026. Holiantispoof: Audio llm for holistic speech anti-spoofing.

Jiangyan Yi, Ye Bai, Jianhua Tao, Haoxin Ma, Zhengkun Tian, Chenglong Wang, Tao Wang, and Ruibo Fu. 2023a. Half-truth: A partially fake audio detection dataset.

Jiangyan Yi, Ruibo Fu, Jianhua Tao, Shuai Nie, Haoxin Ma, Chenglong Wang, Tao Wang, Zhengkun Tian, Xiaohui Zhang, Ye Bai, Cunhang Fan, Shan Liang, Shiming Wang, Shuai Zhang, Xinrui Yan, Le Xu, Zhengqi Wen, Haizhou Li, Zheng Lian, and Bin Liu. 2024. Add 2022: the first audio deep synthesis detection challenge.

Jiangyan Yi, Jianhua Tao, Ruibo Fu, Xinrui Yan, Chenglong Wang, Tao Wang, Chu Yuan Zhang, Xiaohui Zhang, Yan Zhao, Yong Ren, Le Xu, Junzuo Zhou, Hao Gu, Zhengqi Wen, Shan Liang, Zheng Lian, Shuai Nie, and Haizhou Li. 2023b. Add 2023: the second audio deepfake detection challenge.

Lin Zhang, Xin Wang, Erica Cooper, Nicholas Evans, and Junichi Yamagishi. 2023a. The partialspoof database and countermeasures for the detection of short fake speech segments embedded in an utterance. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:813—825.

Lin Zhang, Xin Wang, Erica Cooper, Nicholas Evans, and Junichi Yamagishi. 2023b. Range-based equal error rate for spoof localization.

Lin Zhang, Xin Wang, Erica Cooper, and Junichi Yamagishi. 2021a. Multi-task Learning in Utterance-level and Segmental-level Spoof Detection. In *Proc. 2021 Edition of the Automatic Speaker Verification and Spoofing Countermeasures Challenge*, pages 9–15.

Lin Zhang, Xin Wang, Erica Cooper, Junichi Yamagishi, Jose Patino, and Nicholas Evans. 2021b. An Initial Investigation for Detecting Partially Spoofed Audio. In *Proc. Interspeech 2021*, pages 4264–4268.

Xiao Zhang, Rui Zhao, Junjie Yan, Mengya Gao, Yu Qiao, Xiaogang Wang, and Hongsheng Li. 2019. P2sgrad: Refined gradients for optimizing deep face models.