RTCFake: Speech Deepfake Detection in Real-Time Communication

Jun Xue[1,2], Zhuolin Yi[1,2], Yihuan Huang[1,2], Yanzhen Ren[1,2*], Yujie Chen[4], Cunhang Fan[3]
Zicheng Su[1,2], Yongcheng Zhang[1,2], Bo Cai[1,2]
[1]Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education
[2]School of Cyber Science and Engineering, Wuhan University, Wuhan, China
[3]School of Computer Science and Technology, Anhui University, Hefei, China
[4]Beihang University, Beijing, China
{junxue,yizhuolin,renyz}@whu.edu.cn, cunhang.fan@ahu.edu.cn

Abstract

With the rapid advancement of speech generation technologies, the threat posed by speech deepfakes in real-time communication (RTC) scenarios has intensified. However, existing detection studies mainly focus on offline simulations and struggle to cope with the complex distortions introduced during RTC transmission, including unknown speech enhancement processes (e.g., noise suppression) and codec compression. To address this challenge, we present the first large-scale speech deepfake dataset tailored for RTC scenarios, termed *RTCFake*, totaling approximately 600 hours. The dataset is constructed by transmitting speech through multiple mainstream social media and conferencing platforms (e.g., Zoom), enabling precise pairing between offline and online speech. In addition, we propose a phoneme-guided consistency learning (PCL) strategy that enforces models to learn platform-invariant semantic structural representations. In this paper, the RTCFake dataset is divided into training, development, and evaluation sets. The evaluation set further includes both unseen RTC platforms and unseen complex noise conditions, thereby providing a more realistic and challenging evaluation benchmark for speech deepfake detection. Furthermore, the proposed PCL strategy achieves significant improvements in both cross-platform generalization and noise robustness, offering an effective and generalizable modeling paradigm. The *RTCFake* dataset is provided in the https://huggingface.co/datasets/JunXueTech/RTCFake.

1 Introduction

The rapid evolution of text-to-speech (TTS) (Zhou et al., 2025b) and voice conversion (VC) (Du et al., 2024b) technologies has drastically lowered the barrier for high-fidelity speech synthesis. With the prevalence of online conferencing and remote collaboration, voice has become a cornerstone of identity verification in online interactions. A prime example is a 2025 incident where a corporate executive was targeted in a $499,000 fraud during a Zoom meeting featuring an AI-impersonated CEO (Channel News Asia, 2025). Such deepfake threats have posed a severe risk to the security of Real-Time Communication (RTC).

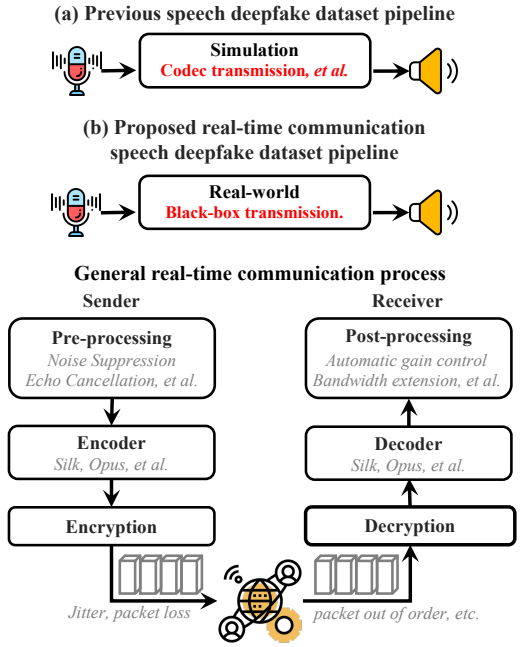

Figure 1: Illustration of different speech deepfake dataset construction pipelines. (a) Previous datasets mainly rely on simulated distortions, such as codec-based transmission, to model communication effects. (b) The proposed *RTCFake* dataset is constructed under real-world black-box transmission conditions, where communication over mainstream social media platforms typically involves key distortion sources, including noise suppression, echo cancellation, codec encoding and decoding, and packet loss during network transmission.

Recently, the research community has introduced a series of Synthetic Speech Detection (SSD) challenges to safeguard the authenticity of voice interactions, gaining significant attention. For instance, ASVspoof 2021 (Yamagishi et al., 2021) and ASVspoof 5 (Wang et al., 2024) highlighted voice distortions caused by codecs and lossy com-

*Corresponding author

Findings of the Association for Computational Linguistics: ACL 2026, pages 5763–5775

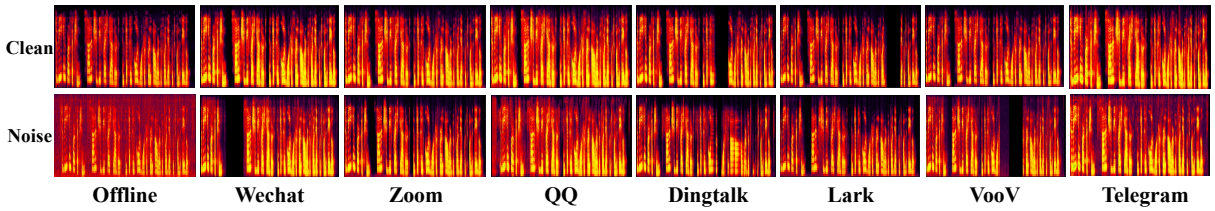

Figure 2: Amplitude spectrograms of the same utterance under offline and multiple online transmission conditions. The top row shows the clean speech, while the bottom row shows the corresponding noisy speech.

pression (e.g., MP3), while ADD 2022 (Yi et al., 2022) incorporated various real-world noises. However, factors in real-world voice interactions are highly coupled and dynamically evolving. Existing challenges often focus on isolated factors, making it difficult to faithfully evaluate model performance in authentic and complex environments.

Mainstream RTC platforms typically exhibit *black-box* characteristics, in which speech undergoes a series of highly integrated processing modules (as illustrated in Fig. 1). On the sender side, speech is first processed by front-end modules such as noise suppression and echo cancellation, followed by encoding and encrypted transmission. During network transmission, the signal is inevitably affected by jitter and packet loss. Finally, on the receiver side, the speech is decrypted, decoded, and further processed through post-processing modules such as gain control and bandwidth extension. These modules are tightly coupled in both the temporal and spectral domains, leading to systematic and nonlinear alterations in the energy distribution and acoustic structure of the transmitted speech (as shown in Fig. 2).

Such realistic and complex RTC environments pose multiple critical challenges for SDD. **(1) Black-box processing:** As the transmission strategies of RTC platforms are unknown, detection models are required to capture forgery cues under black-box conditions. **(2) Noise robustness challenges:** Built-in enhancement algorithms in RTC systems (e.g., noise suppression and echo cancellation), while improving perceptual speech quality, often suppress or distort fine-grained forgery artifacts in synthetic speech. **(3) Cross-platform generalization challenges:** Due to variations in front-end processing logic and transmission protocols across platforms, the distortion distributions imposed on speech signals differ significantly.

To address the above challenges, we introduce the RTCFake dataset, a large-scale speech deepfake dataset specifically constructed for RTC scenarios. The dataset is built by first generating offline data and then transmitting it through mainstream RTC platforms, totalling approximately 600 hours. Based on *RTCFake*, we analyze offline and online speech representations, and observe that phoneme-level representations exhibit substantially higher stability than frame-level representations. Therefore, we propose a phoneme-guided consistency learning strategy, which encourages the model to focus on discriminative features that remain invariant across offline-to-online transformations during training, thereby effectively mitigating the adverse impact of RTC-induced distortions.

The main contributions of this work are summarized as follows:

- ***RTCFake* Dataset:** We construct the first large-scale speech deepfake dataset tailored for RTC scenarios, comprising 600 hours of data transmitted through mainstream communication platforms, which provides a foundation for studying speech deepfake detection under real-world communication conditions.
- **Phoneme-Guided Consistency Learning:** Leveraging the high stability of phoneme-level representations under communication transmission, we propose a phoneme-guided consistency learning strategy that constrains the model to focus on discriminative features that remain invariant across offline–online scenarios during training.
- **Robustness and Generalization Evaluation:** Extensive experiments conducted under multiple noise conditions and cross-platform evaluation settings demonstrate that the proposed dataset and method significantly improve the robustness and generalization performance of detection models in realistic RTC scenarios.

2 Related Work

Speech Deepfake Detection Datasets. In recent years, several key benchmark datasets have

Table 1: Overview of the RTCFake dataset and comparison with existing speech deepfake datasets.

Dataset	Year	Duration*	Description
ASVspoof2019 (Todisco et al., 2019)	2019	116	Clean scenario
ASVspoof2021 (Liu et al., 2023)	2021	–	Codec transmission data
ADD2022 (Yi et al., 2022)	2022	–	Noise scenario
ADD2023 (Yi et al., 2023)	2023	–	Generation–detection adversarial
ASVspoof5 (Wang et al., 2024)	2024	600	Crowdsourced data
CD-ADD (Li et al., 2024b)	2024	384	Cross-domain deepfake
MLAAD (Müller et al., 2024)	2024	–	Multi-language data
DFADD (Du et al., 2024a)	2024	200	Based on diffusion and flow-matching
CVoiceFake (Li et al., 2024a)	2024	–	Content privacy-preserving
FSW (Xie et al., 2025a)	2025	254	Fake speech in-the-wild
SpoofCeleb (Jung et al., 2025)	2025	1982	In the wild
CodecFake (Xie et al., 2025b)	2025	–	Based on audio language model
SpeechFake (Huang et al., 2025)	2025	3000	Multi-language data
RTCFake (Ours)	2025	600	**Real-time Communication**

*Duration is measured in hours.

emerged in the field of SDD. As summarized in Table 1, the ASVspoof series (Todisco et al., 2019; Liu et al., 2023; Wang et al., 2024) established standardized evaluation protocols, with ASVspoof 2021 and ASVspoof 5 incorporating channel codec effects. Subsequently, datasets such as ADD 2023 (Yi et al., 2023), DFADD (Du et al., 2024a), CodecFake (Xie et al., 2025b), and SpeechFake (Huang et al., 2025) introduced high-fidelity synthetic speech based on diffusion models, flow matching, and modern codecs. Furthermore, SpoofCeleb (Jung et al., 2025) and FakeSpeechWild (Xie et al., 2025a) explored the complexities of unconstrained real-world conditions by collecting samples from open platforms.

However, these datasets have not yet accounted for the highly coupled and dynamically evolving black-box processing pipelines inherent in RTC scenarios. To bridge this gap, we construct the RTCFake dataset. By transmitting utterances through mainstream communication platforms to obtain paired "offline-online" speech, RTCFake provides solid data support in speech interaction scenarios.

Speech Deepfake Detection Methods. Existing speech deepfake detection methods primarily focus on mining key discriminative cues from speech signals. Current research typically relies on handcrafted acoustic features (Fan et al., 2024b; Xue et al., 2022), end-to-end deep models (Jung et al., 2022; Xue et al., 2023, 2024), and self-supervised speech representation learning (Zhang et al., 2024). Meanwhile, robust modeling against noise interference (Fan et al., 2024a) and codec-related distortions (Wu et al., 2024) has also received significant attention.

However, most existing methods rely on frame-level acoustic features. In RTC scenarios, these subtle frame-level cues are easily erased by complex nonlinear processing modules. Due to the lack of effective modeling for transmission-invariant features across platforms, existing strategies face severe performance challenges in real-world communication environments.

3 Dataset Collection and Statistics

This section describes the overall construction and transmission pipeline of the *RTCFake* dataset, as illustrated in Fig. 3. The pipeline consists of three successive stages: offline speech construction, noise simulation, and RTC transmission. By integrating diverse speech generation methods, representative environmental disturbances, and end-to-end transmission through mainstream communication platforms, we construct paired offline–online speech data. This dataset provides a solid foundation for conducting SDD research in realistic communication scenarios.

3.1 Speech Construction

The offline dataset is constructed from both real and synthetic speech. Real speech is collected from open-source corpora, with English data sourced

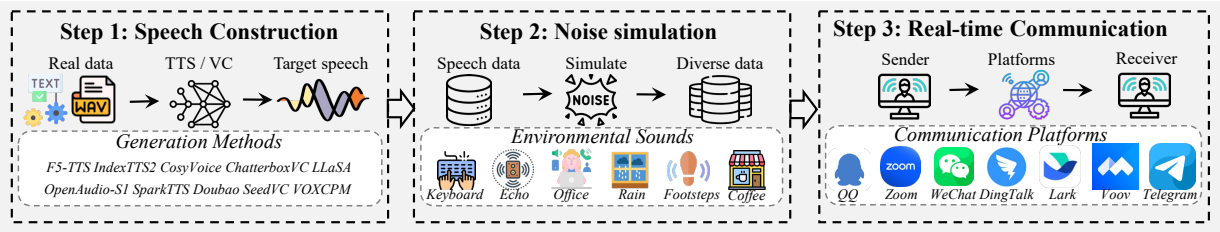

Figure 3: Offline data construction and real-time communication transmission pipeline

from LibriHeavy (Kang et al., 2024) and Chinese data from Chinese-Lips (Zhao et al., 2025), ensuring diversity in natural speech across languages and speaking styles. Synthetic speech is generated using a variety of mainstream TTS and VC systems, as summarized in Table 9. In total, we employ seven TTS systems (G01–G07) and three VC systems (G08–G10), covering a wide range of representative modeling paradigms in contemporary speech generation research.

As shown in Table 6, the offline dataset is divided into training, development, and evaluation subsets. To provide a basis for subsequent analysis of processing mechanisms such as noise suppression and echo cancellation during real-time transmission, multiple scenario-specific noises are introduced into the evaluation set, including office and coffee[1], echo[2], and three types of environmental noise (Piczak, 2015).

3.2 Real-time Communication

To simulate realistic RTC scenarios, we employ two independent PC devices serving as the sender and the receiver, respectively. On the sender side, offline speech samples are played through a virtual audio device and fed into mainstream RTC platforms for voice transmission; on the receiver side, the incoming audio streams are captured and recorded in real time using OBS[3]. The transmission process covers multiple widely used communication platforms, including Wechat, Zoom, QQ, DingTalk, Lark, VooV, and Telegram. To improve transmission efficiency, a subset of speech samples is concatenated prior to transmission; after transmission, the received audio is segmented based on timestamp information to recover individual utterances.

In addition, automatic speech recognition (ASR) (Radford et al., 2023) is used to verify the consistency between the transcribed content of the transmitted speech and the original labels, and samples with mismatched content are discarded. Through this procedure, we obtain online speech samples that closely reflect real user communication behavior while maintaining reliable annotations.

3.3 Dataset Statistics

The RTCFake dataset consists of two subsets, offline and online, with a total duration of approximately 600 hours and coverage of 307 speakers. Table 1 presents a comparison between RTCFake and existing speech deepfake datasets. For a more detailed introduction to the dataset, please refer to Appendix Section A.

4 Methodology

4.1 Analysis and Motivation

The distortions introduced within RTC systems exhibit distinct hierarchical characteristics. Processing modules such as codec compression and noise suppression severely perturb local temporal structures and instantaneous acoustic details, leading to a pronounced distribution shift in frame-level representations between offline and online speech. Nevertheless, these frame-level features often contain critical fine-grained artifacts essential for SDD.

Fig.4 presents a comparison of similarity metrics for frame-level and phoneme-level representations before and after transmission, based on paired offline–online utterances from the RTCFake training set. Statistical results demonstrate that phoneme representations possess significantly higher stability than frame representations, characterized by a larger mean similarity and a substantially smaller variance. This phenomenon suggests that RTC systems prioritize the preservation of semantic intelligibility over acoustic fidelity. Consequently, as the fundamental units of linguistic content, phonemes maintain superior consistency during cross-platform transmission.

[1]https://media.xiph.org/rnnoise/data/
[2]https://github.com/CLAD23/CLAD/tree/main
[3]https://obsproject.com/

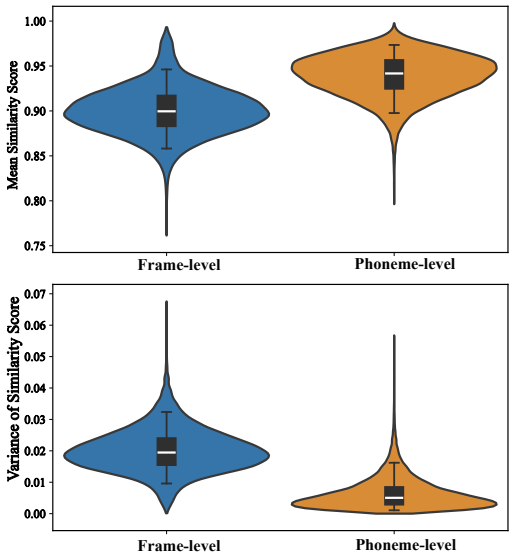

Figure 4: Comparison of offline–online representation similarity at frame-level (Babu et al., 2022) and phoneme-level (Xu et al., 2022) on the RTCFake training set. Phoneme-level representations show higher mean similarity and lower variance, indicating better stability.

Motivated by these findings, we propose a phoneme-guided consistency learning approach during training. Within this framework, phoneme-level representations act as robust anchors for cross-domain modeling, encouraging the neural network to learn RTC platform-invariant feature representations, thereby improving the model's generalization performance.

4.2 Phoneme-guided Consistency Learning

Based on the above observations, we propose a phoneme-guided consistency learning strategy, which aims to enforce invariance between offline and online representations at the level of semantic structural units.

First, the offline and online speech signals are processed by a shared feature extractor to obtain frame-level acoustic representations. Subsequently, a phoneme recognition model[4] is employed to predict frame-level phoneme boundaries, based on which consecutive frames are aligned into linguistically meaningful phoneme segments.

Specifically, let $\boldsymbol{H} = [h_1, h_2, \ldots, h_T]$ denote the sequence of frame-level features. The phoneme-level representation $\boldsymbol{p}_k$ is computed via temporal average pooling:

$$\boldsymbol{p}_k = \frac{1}{|fk|} \sum t \in f_k h_t, \tag{1}$$

where $|f_k|$ denotes the number of frames within the k-th phoneme boundary. This aggregation is applied to both offline and online feature sequences, resulting in paired phoneme-level representations, denoted as $\boldsymbol{p}^{(a)}$ and $\boldsymbol{p}^{(b)}$, respectively.

To enhance representation consistency under different transmission conditions, we introduce a bidirectional consistency constraint between paired phoneme-level features $\boldsymbol{p}^{(a)}$ and $\boldsymbol{p}^{(b)}$. Specifically, consistency is enforced by minimizing the mean squared error (MSE) between the two representations. Accordingly, the phoneme-level consistency learning loss is defined as:

$$\mathcal{L}_{pcl}(\boldsymbol{p}^{(a)}, \boldsymbol{p}^{(b)}) = D_{\text{MSE}}(p^{(a)} \| p^{(b)}), \tag{2}$$

During training, we jointly optimize the classification loss and the phoneme-level consistency learning term to ensure both discriminative capability and representation consistency across different transmission conditions. Specifically, the offline and online branches produce prediction logits, denoted as $\boldsymbol{z}^{(a)}$ and $\boldsymbol{z}^{(b)}$, respectively, each of which is supervised by the ground-truth label $\boldsymbol{y}$ via the cross-entropy loss $\mathcal{L}_{\text{ce}}$.

By combining the above loss terms, the overall training objective is defined as

$$\begin{aligned} \mathcal{L} = \frac{1}{2}\big(\mathcal{L}_{\text{ce}}(\boldsymbol{z}^{(a)}, \boldsymbol{y}) + \mathcal{L}_{\text{ce}}(\boldsymbol{z}^{(b)}, \boldsymbol{y})\big) \\ + \lambda \mathcal{L}_{\text{pcl}}(\boldsymbol{p}^{(a)}, \boldsymbol{p}^{(b)}), \end{aligned} \tag{3}$$

where λ is a weighting coefficient that balances the classification objective and the phoneme-level consistency learning term.

5 Experiments and Analysis

5.1 Experimental Setup

To evaluate deepfake speech detection performance, we adopt a state-of-the-art model, XLSR+AASIST (Tak et al., 2022b). Specifically, XLSR (Babu et al., 2022) serves as the front-end feature extractor, while AASIST employs a heterogeneous stacked graph attention network as the back-end classifier for deepfake detection. During training, we apply RawBoost (Tak et al., 2022a) to improve robustness. For phoneme representation, we employ the pre-trained Wav2Vec2-Large-XLSR-53 model (Xu et al., 2022) to identify phoneme boundaries. The

[4]https://huggingface.co/facebook/wav2vec2-xlsr-53-espeak-cv-ft

Table 2: EER (%) results under different training datasets and evaluation conditions. The best and second-best results in each column are highlighted in bold and underlined, respectively. Off and On denote models trained on offline and online data, respectively, while Mix indicates training on their combination. Details of P0X conditions are provided in Table 5.

Train Data	**Eval Data (EER ↓)**									
	Offline	Online								All
		P01	P02	P03	P04	P05	P06	P07	avg	
ASVspoof2019	51.15	54.68	29.70	49.71	53.87	49.45	48.23	43.67	49.40	50.28
DFADD	47.86	50.45	39.49	46.70	48.38	47.73	48.26	46.42	47.56	47.71
ASVspoof5	42.49	48.76	29.99	44.09	46.86	45.72	44.88	44.66	44.92	43.71
FSW	43.81	38.38	58.19	43.02	40.94	43.52	44.52	43.50	43.55	43.68
CodecFake	42.25	52.90	23.74	39.96	47.55	40.11	43.59	40.33	41.55	41.90
speechfake-BD	35.36	47.48	24.16	38.17	44.00	37.64	38.02	40.38	40.01	37.69
SpoofCeleb	29.56	40.05	30.70	35.16	38.54	36.41	32.48	40.33	38.55	34.06
CD-ADD	33.86	42.13	26.66	32.11	40.32	35.10	34.07	31.57	34.01	33.95
Off	5.42	6.79	20.40	13.10	12.56	16.72	16.07	19.05	13.79	9.60
On	9.57	5.05	7.30	8.05	8.79	10.53	11.77	11.80	8.35	8.96
Mix	6.09	4.93	8.85	8.10	8.53	10.97	11.65	12.18	8.57	7.33
PCL*	**4.84**	**3.79**	**6.24**	**7.03**	**6.76**	**8.51**	**10.17**	**8.75**	**6.77**	**5.81**

*PCL denotes a phoneme-guided consistency learning strategy trained on paired offline–online data.

model is optimized using Adam with a learning rate of 1×10^{-6} and a weight decay of 1×10^{-4}. Training is conducted for up to 100 epochs with an early stopping strategy, where training is terminated if no performance improvement is observed for 10 consecutive epochs. During evaluation, we use the Equal Error Rate (EER) as the performance metric. All experiments are conducted on an NVIDIA RTX 4090 GPU.

5.2 Overall Performance

Table 2 reports EER results under different training datasets and evaluation conditions. The first eight rows correspond to models trained on existing open-source datasets, while the last four rows present models trained on the offline data (Off), online data (On), and mixed data (Mix) constructed in this work, as well as the results obtained after applying the proposed phoneme-guided consistency learning (PCL) strategy.

First, models trained on open-source datasets exhibit very limited generalization capability. They produce extremely high EERs on both the offline test set and multiple online platforms. This indicates that existing open-source datasets fail to capture the complex distributions encountered in real-world applications.

Second, a severe domain mismatch exists between offline and online scenarios. Models trained exclusively on offline data (Off) achieve reasonable performance on the offline test set (5.42%) but suffer dramatic degradation under online conditions, where the average EER increases to 13.79%. This clearly demonstrates that black-box transmission in RTC scenarios poses a substantial challenge to spoofing detection. Models trained exclusively on online data (On) improve robustness in online environments; however, their performance on offline data deteriorates markedly, with the EER rising to 9.57%. This suggests that purely online training introduces domain-specific biases that weaken the model's ability to detect spoofed speech in offline conditions.

Mixed training (Mix) partially alleviates the domain mismatch by jointly modeling offline and online data, resulting in more balanced performance and a reduced average EER of 7.33%. Finally, the proposed PCL method achieves the best overall performance. Compared with the mixed training strategy, PCL consistently reduces EER across all evaluation conditions, achieving the lowest average EER of 5.81%.

In summary: (1) Existing open-source datasets are insufficient for evaluating speech deepfake detection under realistic RTC conditions. (2) RTC transmission induces a severe domain mismatch between offline and online scenarios. (3) Phoneme-guided consistency learning (PCL) effectively exploits cross-scenario stable characteristics and substantially mitigates domain mismatch under RTC

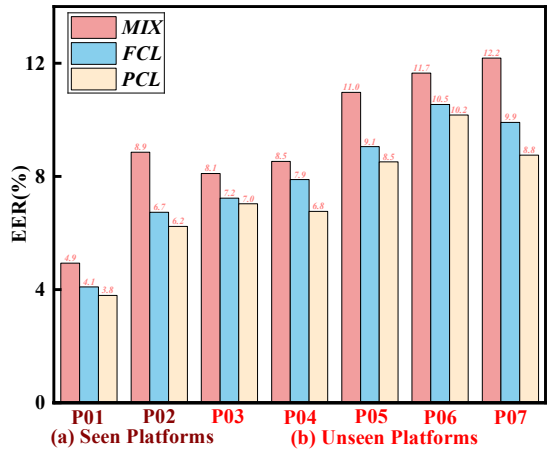

Figure 5: Performance comparison of mix training (MIX), frame-level consistency learning (FCL), and phoneme-level consistency learning (PCL) across seen and unseen communication platforms.

conditions.

5.3 Cross-Platform Generalization

Fig. 5 presents a comparison of cross-platform generalization performance, where (a) and (b) correspond to platforms seen and unseen during training, respectively. Three training strategies are compared: a baseline using mixed offline and online data (MIX), frame-level consistency learning (FCL), and phoneme-level consistency learning (PCL). This experiment is designed to evaluate the impact of different consistency modeling granularities on model generalization under cross-platform conditions.

As shown in Fig. 5, both FCL and PCL consistently outperform the MIX baseline on seen platforms, demonstrating the effectiveness of consistency learning in mitigating the distribution discrepancy between offline and online data. Notably, PCL achieves lower EERs than FCL across all seen platforms, substantiating the prior analysis that phoneme-level modeling provides more stable representation alignment than frame-level constraints.

Under unseen platform conditions, the performance gap among the methods becomes even more pronounced. The MIX strategy suffers from severe performance degradation, particularly on platforms P05–P07, where EERs increase significantly. While FCL alleviates this degradation to some extent, it still exhibits notable platform-wise variability. In contrast, PCL maintains the lowest and most stable EERs across all unseen platforms. Its advantages are particularly evident on platforms with more severe distortions (e.g., P07). These results suggest that by mining robust features within linguistic structures, phoneme-level consistency learning enables the model to learn platform-invariant representations, thereby effectively enhancing generalization in real-world, complex communication environments.

Table 3: EER (%) results under different evaluation conditions, averaged over offline and online settings. Details of S0X conditions are provided in Table 5.

Train	Seen	Unseen					
	S01*	S02	S03	S04	S05	S06	S07
Off	7.68	17.24	16.05	16.56	18.65	14.28	15.28
On	6.66	12.33	12.60	17.34	14.30	11.27	11.92
Mix	5.63	12.80	12.72	16.92	13.61	12.11	10.80
PCL	**3.88**	**10.95**	**9.30**	**13.40**	**13.09**	**9.57**	**9.53**

*S01 denotes clean-only.

5.4 Robustness under Noise Scenarios

Table 3 evaluates the robustness of different models under unseen noise conditions, where S01 corresponds to the clean condition and S02–S07 represent various types of unseen noise and interference scenarios. The results indicate that performance differences among training strategies become more pronounced under these unseen conditions, reflecting varying levels of generalization capability to complex real-world environments. It is worth noting that, in RTC scenarios, speech degradation is not only caused by environmental noise itself but is also significantly influenced by built-in speech enhancement modules of communication platforms, such as noise suppression, echo cancellation, and automatic gain control. The coupling between noise and enhancement processing introduces more complex and nonlinear distortion patterns.

By comparing single-source training strategies (Off / On) with mixed-data training (Mix), we observe that simple data mixing can partially alleviate distribution shifts under unseen conditions, leading to lower EERs in some noise scenarios. However, these improvements are not consistent across different unseen noise conditions, suggesting that relying solely on data-level mixing is insufficient to effectively model the complex distortion responses introduced by speech enhancement in real-time communication systems.

In contrast, the PCL method demonstrates more

Table 4: Ablation study of feature granularity and training strategies, where EER results are averaged over offline and online evaluation settings.

Feature		Strategy		EER (%)
Frame	Phoneme	FCL	PCL	
✗	✓	✓	✗	8.34
✗	✓	✗	✓	7.52
✓	✗	✓	✗	6.55
✓	✗	✗	✓	**5.81**

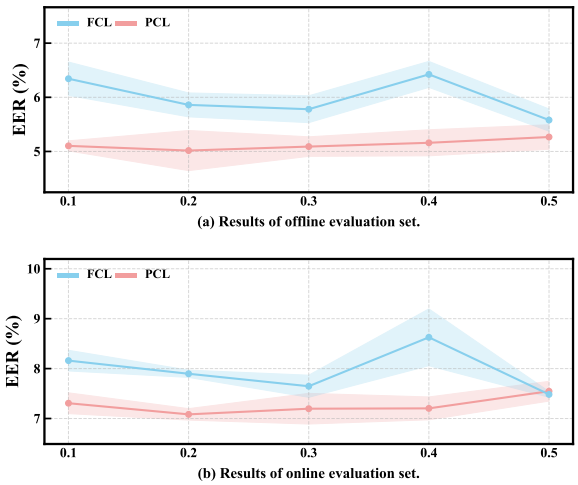

Figure 6: Comparison of FCL and PCL consistency learning on offline and online evaluation sets. Colored curves denote the mean EER across multiple runs under different values of the regularization weight λ, while the shaded regions indicate the corresponding minimum–maximum range.

stable performance across unseen noise scenarios. By enforcing consistency between offline and online representations at the phoneme level, PCL effectively suppresses structural shifts induced by speech enhancement modules, achieving a better balance between environmental noise variations and platform-specific processing distortions, and thereby significantly improving robustness under unseen real-time communication noise conditions.

5.5 Ablation Studies

In this subsection, we conduct ablation studies to validate the effectiveness of the proposed method.

Feature and Training Strategy Analysis. As shown in Table 4, we compare different combinations of feature granularity and consistency strategies. The results indicate that frame-level features provide a stronger performance baseline than phoneme-level features. From the perspective of training strategies, PCL significantly outperforms frame-level constraints. These findings suggest that preserving fine-grained low-level features while leveraging PCL to capture structured semantic information is more effective for RTC scenarios.

Weight and Stability Analysis. Based on the above observations, Fig. 6 further investigates the stability of different consistency strategies under varying weighting factors λ. Overall, PCL consistently outperforms FCL across all evaluation sets. Notably, the performance of PCL exhibits smaller fluctuations with respect to changes in λ, and the variance of the experimental results is substantially reduced. This demonstrates that phoneme-level constraints provide a more stable and robust regularization signal.

6 Conclusion

This paper presents RTCFake, a pioneering speech deepfake detection dataset specifically constructed to investigate black-box transmission conditions within RTC scenarios. Through in-depth analysis, we reveal that while the coupled nonlinear processing in RTC systems severely distorts fine-grained acoustic details, phoneme-level representations maintain superior structural stability. Motivated by this, we propose a phoneme-guided consistency learning method that enforces alignment between offline and online representations at the semantic structural level, effectively mitigating the loss of discriminative cues during transmission. Extensive experiments demonstrate that our approach significantly enhances detection robustness and generalization across both seen and unseen communication platforms, as well as under diverse noise conditions. This work provides a solid data foundation and a robust methodological framework for deploying speech deepfake detection systems in realistic and complex communication environments.

7 Acknowledgement

This work is supported by the Natural Science Foundation of China (NSFC) under the grant NO.62572358, 62571002

Limitations

Despite constructing the RTCFake dataset and proposing the phoneme-guided consistency learning method, this work has several limitations. First,

real-world communication involves a vast array of confounding factors beyond transmission, such as the heterogeneity of recording/playback hardware and diverse user behaviors. The complex interplay between these terminal-side variables and the platforms' internal processing pipelines may introduce additional signal perturbations not fully captured in this study. Moreover, while our phoneme-level constraints demonstrate superior stability, a performance gap still exists when encountering extreme unseen noise or highly aggressive nonlinear distortions in certain communication platforms. Future research will focus on developing platform-agnostic and more adaptive representation modeling to further bridge the gap between laboratory evaluation and large-scale real-world deployment.

Ethics Statement

The real speech samples used in the RTCFake dataset are sourced from publicly available speech corpora that are commonly employed in speech processing research. Based on these samples, deepfake speech is generated using text-to-speech (TTS) and voice conversion (VC) techniques, and subsequently transmitted through mainstream real-time communication platforms to construct online speech samples. The dataset does not include speech from identifiable real individuals, nor does it contain any harmful, sensitive, or privacy-related content.

Generative AI Use Disclosure

Generative AI tools were used exclusively for language refinement and grammatical correction during the preparation of this manuscript.

References

2025. https://www.volcengine.com/docs/6561/1257584?lang=en.

Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, and 1 others. 2022. Xls-r: Self-supervised cross-lingual speech representation learning at scale. In *Proc. Interspeech 2022*, pages 2278–2282.

Channel News Asia. 2025. Company finance director nearly loses over US$499,000 to scammers using deepfake to impersonate ceo. Accessed: 2025-11-6.

Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, JianZhao JianZhao, Kai Yu, and Xie Chen. 2025. F5-tts: A fairytaler that fakes fluent and faithful speech with flow matching. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6255–6271.

Jiawei Du, I-Ming Lin, I-Hsiang Chiu, Xuanjun Chen, Haibin Wu, Wenze Ren, Yu Tsao, Hung-yi Lee, and Jyh-Shing Roger Jang. 2024a. Dfadd: The diffusion and flow-matching based audio deepfake dataset. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pages 921–928. IEEE.

Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, and 1 others. 2024b. Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens. *arXiv preprint arXiv:2407.05407*.

Cunhang Fan, Mingming Ding, Jianhua Tao, Ruibo Fu, Jiangyan Yi, Zhengqi Wen, and Zhao Lv. 2024a. Dual-branch knowledge distillation for noise-robust synthetic speech detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:2453–2466.

Cunhang Fan, Jun Xue, Jianhua Tao, Jiangyan Yi, Chenglong Wang, Chengshi Zheng, and Zhao Lv. 2024b. Spatial reconstructed local attention res2net with f0 subband for fake speech detection. *Neural Networks*, 175:106320.

Wen Huang, Yanmei Gu, Zhiming Wang, Huijia Zhu, and Yanmin Qian. 2025. Speechfake: A large-scale multilingual speech deepfake dataset incorporating cutting-edge generation methods. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, page 9985–9998. Association for Computational Linguistics.

Jee-weon Jung, Hee-Soo Heo, Hemlata Tak, Hye-jin Shim, Joon Son Chung, Bong-Jin Lee, Ha-Jin Yu, and Nicholas Evans. 2022. Aasist: Audio anti-spoofing using integrated spectro-temporal graph attention networks. In *ICASSP 2022-2022 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 6367–6371. IEEE.

Jee-weon Jung, Yihan Wu, Xin Wang, Ji-Hoon Kim, Soumi Maiti, Yuta Matsunaga, Hye-jin Shim, Jinchuan Tian, Nicholas Evans, Joon Son Chung, and 1 others. 2025. Spoofceleb: Speech deepfake detection and sasv in the wild. *IEEE Open Journal of Signal Processing*.

Wei Kang, Xiaoyu Yang, Zengwei Yao, Fangjun Kuang, Yifan Yang, Liyong Guo, Long Lin, and Daniel Povey. 2024. Libriheavy: A 50,000 hours asr corpus with punctuation casing and context. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10991–10995. IEEE.

Xinfeng Li, Kai Li, Yifan Zheng, Chen Yan, Xiaoyu Ji, and Wenyuan Xu. 2024a. Safeear: Content privacy-preserving audio deepfake detection. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*, pages 3585–3599.

Yuang Li, Min Zhang, Mengxin Ren, Xiaosong Qiao, Miaomiao Ma, Daimeng Wei, and Hao Yang. 2024b. Cross-domain audio deepfake detection: Dataset and analysis. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4977–4983.

Shijia Liao, Yuxuan Wang, Tianyu Li, Yifan Cheng, Ruoyi Zhang, Rongzhi Zhou, and Yijin Xing. 2024. Fish-speech: Leveraging large language models for advanced multilingual text-to-speech synthesis. *arXiv preprint arXiv:2411.01156*.

Songting Liu. 2024. Zero-shot voice conversion with diffusion transformers. *arXiv preprint arXiv:2411.09943*.

Xuechen Liu, Xin Wang, Md Sahidullah, Jose Patino, Héctor Delgado, Tomi Kinnunen, Massimiliano Todisco, Junichi Yamagishi, Nicholas Evans, Andreas Nautsch, and 1 others. 2023. Asvspoof 2021: Towards spoofed and deepfake speech detection in the wild. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:2507–2522.

Nicolas M Müller, Piotr Kawa, Wei Herng Choong, Edresson Casanova, Eren Gölge, Thorsten Müller, Piotr Syga, Philip Sperl, and Konstantin Böttinger. 2024. Mlaad: The multi-language audio anti-spoofing dataset. In *2024 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE.

Karol J Piczak. 2015. Esc: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 1015–1018.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.

Resemble AI. 2025. Chatterbox-TTS. https://github.com/resemble-ai/chatterbox. GitHub repository.

Hemlata Tak, Madhu Kamble, Jose Patino, Massimiliano Todisco, and Nicholas Evans. 2022a. Rawboost: A raw data boosting and augmentation method applied to automatic speaker verification anti-spoofing. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6382–6386. IEEE.

Hemlata Tak, Massimiliano Todisco, Xin Wang, Jeeweon Jung, Junichi Yamagishi, and Nicholas Evans. 2022b. Automatic speaker verification spoofing and deepfake detection using wav2vec 2.0 and data augmentation. In *The Speaker and Language Recognition Workshop (Odyssey 2022)*. ISCA.

Massimiliano Todisco, Xin Wang, Ville Vestman, Md Sahidullah, Hector Delgado, Andreas Nautsch, Junichi Yamagishi, Nicholas Evans, Tomi Kinnunen, and Kong Aik Lee. 2019. Asvspoof 2019: Future horizons in spoofed and fake audio detection. In *Interspeech 2019*, pages 1008–1012. International Speech Communication Association.

Xin Wang, Héctor Delgado, Hemlata Tak, Jee-Weon Jung, Hye-Jin Shim, Massimiliano Todisco, Ivan Kukanov, Xuechen Liu, Md Sahidullah, Tomi Kinnunen, and 1 others. 2024. Asvspoof 5: crowdsourced speech data, deepfakes, and adversarial attacks at scale. In *The Automatic Speaker Verification Spoofing Countermeasures Workshop (ASVspoof 2024)*, pages 1–8. ISCA.

Xinsheng Wang, Mingqi Jiang, Ziyang Ma, Ziyu Zhang, Songxiang Liu, Linqin Li, Zheng Liang, Qixi Zheng, Rui Wang, Xiaoqin Feng, and 1 others. 2025. Spark-tts: An efficient llm-based text-to-speech model with single-stream decoupled speech tokens. *arXiv preprint arXiv:2503.01710*.

Haibin Wu, Yuan Tseng, and Hung-yi Lee. 2024. Codecfake: Enhancing anti-spoofing models against deepfake audios from codec-based speech synthesis systems. In *Proc. Interspeech 2024*, pages 1770–1774.

Yuankun Xie, Ruibo Fu, Xiaopeng Wang, Zhiyong Wang, Ya Li, Zhengqi Wen, Haonnan Cheng, and Long Ye. 2025a. Fake speech wild: Detecting deepfake speech on social media platform. *arXiv preprint arXiv:2508.10559*.

Yuankun Xie, Yi Lu, Ruibo Fu, Zhengqi Wen, Zhiyong Wang, Jianhua Tao, Xin Qi, Xiaopeng Wang, Yukun Liu, Haonan Cheng, and 1 others. 2025b. The codecfake dataset and countermeasures for the universally detection of deepfake audio. *IEEE Transactions on Audio, Speech and Language Processing*.

Qiantong Xu, Alexei Baevski, and Michael Auli. 2022. Simple and effective zero-shot cross-lingual phoneme recognition. In *Proc. Interspeech 2022*, pages 2113–2117.

Jun Xue, Cunhang Fan, Zhao Lv, Jianhua Tao, Jiangyan Yi, Chengshi Zheng, Zhengqi Wen, Minmin Yuan, and Shegang Shao. 2022. Audio deepfake detection based on a combination of f0 information and real plus imaginary spectrogram features. In *Proceedings of the 1st international workshop on deepfake detection for audio multimedia*, pages 19–26.

Jun Xue, Cunhang Fan, Jiangyan Yi, Chenglong Wang, Zhengqi Wen, Dan Zhang, and Zhao Lv. 2023. Learning from yourself: A self-distillation method for fake speech detection. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Jun Xue, Cunhang Fan, Jiangyan Yi, Jian Zhou, and Zhao Lv. 2024. Dynamic ensemble teacher-student distillation framework for light-weight fake audio detection. *IEEE Signal Processing Letters*, 31:2305–2309.

Junichi Yamagishi, Xin Wang, Massimiliano Todisco, Md Sahidullah, Jose Patino, Andreas Nautsch, Xuechen Liu, Kong Aik Lee, Tomi Kinnunen, Nicholas Evans, and 1 others. 2021. Asvspoof 2021: accelerating progress in spoofed and deepfake speech detection. In *Proc. ASVSPOOF 2021*, pages 47–54.

Zhen Ye, Xinfa Zhu, Chi-Min Chan, Xinsheng Wang, Xu Tan, Jiahe Lei, Yi Peng, Haohe Liu, Yizhu Jin, Zheqi Dai, and 1 others. 2025. Llasa: Scaling train-time and inference-time compute for llama-based speech synthesis. *arXiv preprint arXiv:2502.04128*.

Jiangyan Yi, Ruibo Fu, Jianhua Tao, Shuai Nie, Haoxin Ma, Chenglong Wang, Tao Wang, Zhengkun Tian, Ye Bai, Cunhang Fan, and 1 others. 2022. Add 2022: the first audio deep synthesis detection challenge. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 9216–9220. IEEE.

Jiangyan Yi, Jianhua Tao, Ruibo Fu, Xinrui Yan, Chenglong Wang, Tao Wang, Chu Yuan Zhang, Xiaohui Zhang, Yan Zhao, Yong Ren, and 1 others. 2023. Add 2023: the second audio deepfake detection challenge. *arXiv preprint arXiv:2305.13774*.

Qishan Zhang, Shuangbing Wen, and Tao Hu. 2024. Audio deepfake detection with self-supervised xls-r and sls classifier. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 6765–6773.

Jinghua Zhao, Yuhang Jia, Shiyao Wang, Jiaming Zhou, Hui Wang, and Yong Qin. 2025. Chinese-lips: A chinese audio-visual speech recognition dataset with lip-reading and presentation slides. *arXiv preprint arXiv:2504.15066*.

Siyi Zhou, Yiquan Zhou, Yi He, Xun Zhou, Jinchao Wang, Wei Deng, and Jingchen Shu. 2025a. Indextts2: A breakthrough in emotionally expressive and duration-controlled auto-regressive zero-shot text-to-speech. *arXiv preprint arXiv:2506.21619*.

Yixuan Zhou, Guoyang Zeng, Xin Liu, Xiang Li, Renjie Yu, Ziyang Wang, Runchuan Ye, Weiyue Sun, Jiancheng Gui, Kehan Li, and 1 others. 2025b. Voxcpm: Tokenizer-free tts for context-aware speech generation and true-to-life voice cloning. *arXiv preprint arXiv:2509.24650*.

A Dataset Details

A.1 Dataset Partition

The construction and partition of the dataset in this study involve multiple speech generation models, RTC platforms, and noise conditions. To facilitate a unified description of the experimental setup, simplify subsequent statistical analysis and result reporting, and avoid repetitive and lengthy naming in tables and the main text, we assign unified identifiers to these components and perform systematic dataset partitioning and statistics based on this encoding scheme.

Table 5 summarizes the overall configuration of the speech generation methods, transmission platforms, and noise conditions used in this work. Specifically, different speech generation models, including TTS and VC approaches, are denoted as G01–G10; mainstream real-time communication platforms are denoted as P01–P07; and different noise or interference scenarios are denoted as S01–S07. This identifier scheme is consistently used throughout the experimental setup, dataset partition description, and result analysis to improve conciseness and readability.

Based on this configuration, Table 6 presents the partitioning setup and speaker gender statistics of the RTCFake dataset across the training (Train), validation (Dev), and evaluation (Eval) subsets. For each subset, the included ranges of speech generation models (denoted by Gxx identifiers) and the corresponding platform ranges (Pxx) are explicitly specified, together with the numbers of female and male speakers. This table aims to provide an intuitive comparison of the composition and scale differences among subsets of the offline dataset from the perspectives of speaker distribution and generation method coverage.

Furthermore, Table 7 provides a comprehensive statistical summary of the sample counts for both bonafide and fake speech under offline and online settings. We report the number of samples for the training, development, and evaluation phases, along with their respective totals. These statistics reveal the scale, class imbalance, and distribution characteristics of the data under different environmental constraints, providing essential context for interpreting the performance benchmarks in the following sections.

Overall, through the joint presentation of these Tables, we systematically characterize the construction elements, partitioning strategies, and scale distribution of our dataset, thereby providing a solid foundation for rigorous result analysis.

A.2 Dataset Metadata

RTCFake provides comprehensive metadata for each speech sample to support flexible dataset partitioning, experimental analysis, and reproducibility.

Table 5: Configuration of speech generation methods, transmission platforms, and noise conditions.

ID	Type	Generation	ID	Platform	ID	Noise
G01	TTS	F5-TTS (Chen et al., 2025)	P01	Zoom	S01	Clean
G02	TTS	OpenAudio-S1 (Liao et al., 2024)	P02	QQ	S02	Office
G03	TTS	VOXCPM (Zhou et al., 2025b)	P03	Wechat	S03	Coffee
G04	TTS	LLaSA (Ye et al., 2025)	P04	Dingtalk	S04	Echo
G05	TTS	IndexTTS2 (Zhou et al., 2025a)	P05	Lark	S05	Rain
G06	TTS	Doubao (vol, 2025)	P06	Voov	S06	Footsteps
G07	TTS	SparkTTS (Wang et al., 2025)	P07	Telegram	S07	Keyboard
G08	VC	CosyVoice (Du et al., 2024b)	–	–	–	–
G09	VC	SeedVC (Liu, 2024)	–	–	–	–
G10	VC	ChatterboxVC (Resemble AI, 2025)	–	–	–	–

Table 6: Subset configuration and speaker gender statistics of the RTCFake dataset.

Subset	#Gen	#Platform	#Female	#Male
Train	G01–G04 G08–G09	P01–P02	55	50
Dev	G01–G04 G08–G09	P01–P03	10	11
Eval	G01–G07 G08–G10	P01–P07	95	86

Table 7: Statistics of real and fake samples under offline and online settings.

Set	Train	Dev	Eval	Total
Real Data				
offline	7463	1452	14948	23863
online	6890	1397	22737	31024
Fake Data				
offline	31197	6287	74806	112290
online	30235	6068	113596	149899

Table 8: Training configurations of the W2V+ AASIST model used in experiments.

Configurations	W2V+ AASIST
Model Size	3M
Input Audio	16K
Data augmentation	Rawboost
Optimizer	Adam
Learning Rate	1e-6
Weight Decay	1e-4
Total Epochs	100
Early Stopping	10 epochs
Loss Function	Cross Entropy
Constraint Loss	Mean Squared Error

The metadata associated with each sample includes the following components:

- **Basic Labels:** Indicating whether the speech sample is real or fake.
- **Generation Method:** Specifying the speech generation approach used to create the sample, including TTS and VC models. Each generation method is represented using a unified identifier (e.g., G01–G10).
- **RTC Platform:** Samples in the online subset are annotated with the RTC platform through which they are transmitted, such as Zoom, QQ, or WeChat. Different platforms are encoded using platform identifiers (e.g., P01–P07).
- **Speaker ID:** Providing a unique identity label for the speaker associated with the speech sample. For spoofed samples, the speaker ID corresponds to the target or source speaker used during speech generation.
- **Language ID:** Indicating the language of the audio sample.
- **Text Description:** Providing the corresponding textual transcription of the speech content, which serves as the input text for TTS generation or the reference content for speech analysis.

A.3 Generation and Platforms

Table 9 summarizes the 10 speech generation tools and 7 mainstream RTC platforms used to construct the RTCFake dataset. These tools represent the

Table 9: Speech generation models and real-time communication (RTC) platforms used in this work.

No.	Generation	Link
1	F5-TTS (Chen et al., 2025)	https://github.com/SWivid/F5-TTS
2	OpenAudio-S1 (Liao et al., 2024)	https://github.com/fishaudio/fish-speech
3	VOXCPM (Zhou et al., 2025b)	https://github.com/OpenBMB/VoxCPM
4	LLaSA (Ye et al., 2025)	https://huggingface.co/HKUSTAudio/Llasa-3B
5	IndexTTS2 (Zhou et al., 2025a)	https://github.com/index-tts/index-tts?tab=readme-ov-file
6	Doubao (vol, 2025)	https://www.volcengine.com/docs/6561/1257584?lang=en
7	SparkTTS (Wang et al., 2025)	https://github.com/SparkAudio/Spark-TTS
8	CosyVoice (Du et al., 2024b)	https://github.com/FunAudioLLM/CosyVoice
9	SeedVC (Liu, 2024)	https://github.com/Plachtaa/seed-vc
10	ChatterboxVC (Resemble AI, 2025)	https://github.com/resemble-ai/chatterbox

No.	Platform	Link
1	Zoom	https://www.zoom.com/
2	QQ	https://im.qq.com/index/
3	WeChat	https://www.wechat.com/
4	DingTalk	https://www.dingtalk.com/en
5	Lark	https://www.larksuite.com/en_sg/
6	VooV	https://voovmeeting.com/
7	Telegram	https://telegram.org/

state-of-the-art (SOTA) in both TTS and VC. Furthermore, the selected RTC platforms are widely utilized in social, professional, and educational scenarios, ensuring the practical relevance of the dataset.

B Experiment Details

B.1 Experiment Setting

Table 8 summarizes the training configuration of the W2V+ AASIST (Tak et al., 2022b) model used in our experiments. We follow the standard training setup adopted in prior AASIST-based speech deepfake detection studies. The model takes 16 kHz audio as input and is optimized using the Adam optimizer with a learning rate of 1×10^{-6} and a weight decay of 1×10^{-4}, with RawBoost (Tak et al., 2022a) applied as the data augmentation strategy. The model is trained for up to 100 epochs, with early stopping triggered if no improvement on the validation set is observed for 10 consecutive epochs. During training, cross-entropy loss is used for the classification objective, while mean squared error loss is employed for consistency learning.