

Towards Generalized Source Tracing for Codec-Based Deepfake Speech

Xuanjun Chen^{1*}, I-Ming Lin^{1*}, Lin Zhang², Haibin Wu^{3†}, Hung-yi Lee¹, Jyh-Shing Roger Jang¹
¹National Taiwan University ²Johns Hopkins University ³Independent Researcher

Abstract—Recent attempts at source tracing for codec-based deepfake speech (CodecFake), generated by neural audio codec-based speech generation (CoSG) models, have exhibited suboptimal performance. However, how to train source tracing models using simulated CoSG data while maintaining strong performance on real CoSG-generated audio remains an open challenge. In this paper, we show that models trained solely on codec-resynthesized data tend to overfit to non-speech regions and struggle to generalize to unseen content. To mitigate these challenges, we introduce the Semantic-Acoustic Source Tracing Network (SASTNet), which jointly leverages Whisper for semantic feature encoding and Wav2vec2 with AudioMAE for acoustic feature encoding. Our proposed SASTNet achieves state-of-the-art performance on the CoSG test set of *CodecFake+* dataset, demonstrating its effectiveness for reliable source tracing.

Index Terms—Anti-spoofing, source tracing, audio deepfake detection, neural audio codec, explainability

I. INTRODUCTION

Deepfake detection determines whether the given speech is a bona fide speech or a deepfake speech. Numerous challenges and datasets [1, 2] have been introduced, such as ASVspoof [3–7] and ADD [8, 9]. Recently, attention has shifted from merely detecting deepfake speech to tracing its source. Source tracing offers important insights into the origin or generation algorithm of a deepfake speech, which is especially valuable in out-of-domain detection scenarios. For example, Track 3 of ADD 2023 [9] focused explicitly on identifying the underlying algorithms of deepfake utterances, and Interspeech 2025 is hosting a special session¹ about deepfake source tracing.

The simplest approach to source tracing [10] involves mapping deepfake utterances back to the specific attack model ID that generated them [9]. To enhance generalization, the field has shifted toward attribute-based source tracing [11–14]. Rather than treating each deepfake system as a unique category, this method groups systems according to shared attributes, reflecting the fact that many deepfake pipelines reuse common components. By focusing on these attributes, models can more effectively trace spoofing algorithms that were unseen during training but are built from known elements during training [14]. Drawing on well-defined properties from traditional text-to-speech (TTS) and voice conversion (VC) architectures, researchers have concentrated on classifying attributes such as input types [14], acoustic models [12, 14], speaker representations [13], and vocoders [11–14], etc.

However, these prior studies target source tracing in traditional TTS/VC systems, and only initial efforts have explored codec-based speech generation (CoSG) [15] models. Xie et al. [16] treat source tracing as a closed-set classification problem—identifying known CoSG system IDs and using a binary out-of-distribution detector, but this approach fails when multiple unseen generators are encountered. Although there has been a series of recent works on source tracing [17–19], few focus on codec-based deepfake speech [20, 21]. More recently, Chen et al. [22] utilize the organized *CodecFake+* [21] taxonomy, which covers vector quantization schemes, auxiliary objectives, and decoder types, to formulate three source-tracing tasks grounded in codec attributes. They [22] also show that a model trained on codec-resynthesized data suffers from severe generalization issues when evaluated on CoSG-generated speech, driven by unseen codec types.

This paper further demonstrates that a model trained on codec re-synthesis speech tends to overfit to non-speech regions and unseen speech content. To address these challenges, we propose a Semantic-Acoustic Source Tracing Network (SASTNet) that leverages a semantic Encoder, a coarse-to-fine acoustic Encoder that jointly preserves content and captures fine-grained codec signatures, resulting in significant improvements in codec-based deepfake source tracing on *CodecFake+* [21]. Codebase will be released to support future research ².

II. BACKGROUND

Neural audio codecs [15, 23–26] leave “fingerprints” across three axes: vector quantization, auxiliary objectives, and decoder type. Together, these axes span most codec designs and enable robust tracing of codec-based deepfake. Building on the taxonomy, Chen et al. [22] define three multi-class classification tasks for CodecFake source tracing:

- **Vector Quantization Classification (VQ Task):** Four-class classification: classifies a waveform as real or generated by a model with codec using multi-codebook, single-codebook, or scalar quantization.
- **Auxiliary Objective Classification (AUX Task):** Three-class classification: classifies a waveform as either real or generated by a model with a codec using semantic distillation, or disentanglement objectives.
- **Decoder Type Classification (DEC Task):** Three-class classification: classifies an input utterance as real or generated by a model with codec employing a time- or frequency-domain decoder.

*Equal Contribution. † Corresponding Author

¹<https://deepfake-total.com/sourcetracing>

²<https://github.com/ResponsibleGenAI/CodecFake-Source-Tracing>

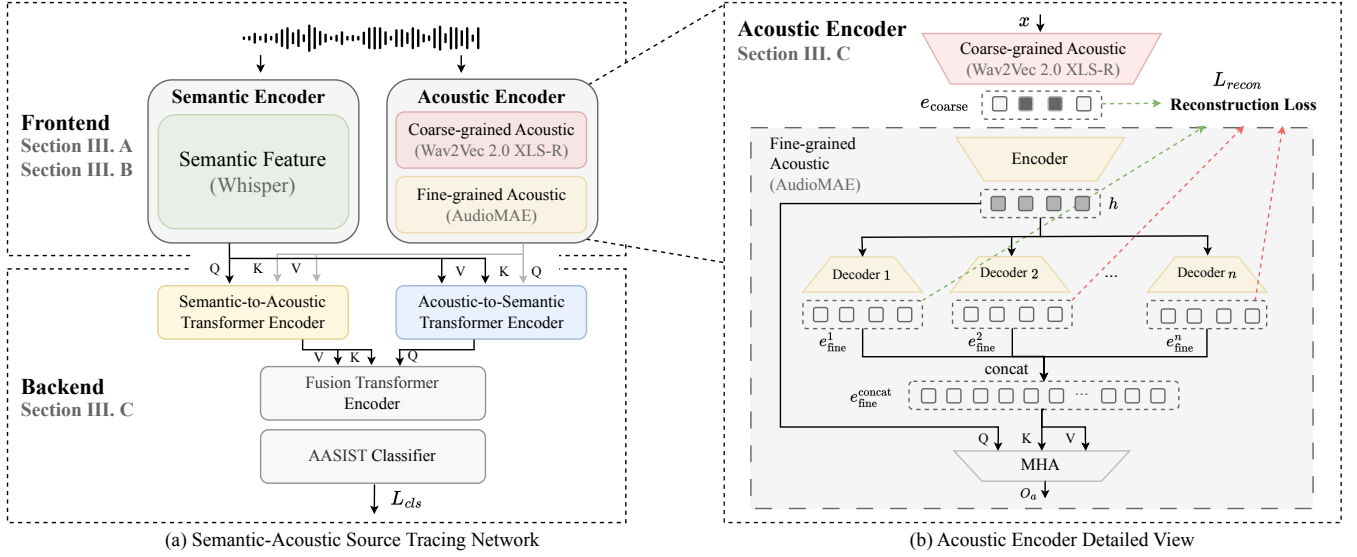


Fig. 1. The Overview of Semantic-Acoustic Source Tracing Network.

Additionally, a traditional anti-spoofing task is included:

- **Binary Spoof Detection (BIN Task):** Classifies each input utterance as bona fide or spoofed speech.

III. SEMANTIC-ACOUSTIC SOURCE TRACING NETWORK

In this section, we introduce our Semantic-Acoustic Source Tracing Network (SASTNet) and explain the intuition behind its design. Chen et al. [22] previously applied a self-supervised learning model as a front-end feature extractor, followed by a classifier for source tracing, but their approach often emphasizes acoustic information at the expense of semantic cues. In contrast, we recognize that effective source tracing requires both the fine-grained acoustic patterns introduced by codec distortions and the preservation of semantic consistency. Indeed, Kawa et al. [27] have shown that combining semantic and acoustic features yields better capture of deepfake artifacts. Because relying solely on acoustic cues from codec distortions overlooks a critical fact: even codec-unprocessed speech exhibits spectral and temporal differences when the spoken content changes. If we focus only on acoustic features and ignore linguistic content, natural variations caused by different words or pronunciations can be mistaken for codec artifacts. In contrast, by preserving semantic information, the network can focus on the content-irrelevant information of the input utterance and then isolate only those subtle acoustic deviations that are independent of what is being said.

Figure 1-(a) shows the framework of SASTNet. Given an input waveform, the model predicts the codec source label based on the tasks defined in Section II. SASTNet has three components: two front-end encoders and a back-end classifier. Front-end encoders include a Semantic Encoder, which extracts high-level linguistic embeddings, and a Coarse-to-Fine Acoustic Encoder, which captures both coarse-grained and fine-grained codec-specific acoustic features. The back-end classifier is a component of three two-layer cross-modal Transformers, Semantic-to-Acoustic (SA), Acoustic-to-Semantic (AS), and Fusion Transformer, each built around a

cross-attention mechanism. In the SA Transformer, semantic features serve as queries while acoustic features act as keys and values; the AS Transformer swaps these roles. The Fusion Transformer then takes AS outputs as queries and SA outputs as keys and values to integrate both modalities. Finally, an AASIST [28] backend model uses the fused representation to produce the source-tracing prediction. The detailed settings of each block are described in the following sections.

A. Semantic Encoder

The Semantic Encoder leverages a frozen, pre-trained Whisper [29] model to generate content-robust embeddings that serve as a stable anchor for source tracing. By verifying that two utterances share identical linguistic content, these embeddings ensure the network ignores natural acoustic variations caused by different words or pronunciations and focuses only on codec-induced distortions. Concretely, let x denote the raw input waveform. We first zero-pad x to a fixed duration of 30 seconds, yielding x_{pad} . We then pass x_{pad} through Whisper’s encoder to obtain a feature tensor $s_{\text{pad}} \in \mathbb{R}^{1500 \times 768}$. Because only the initial tokens cover the same time span as our acoustic features, we truncate s_{pad} to the first 256 non-padding time steps: $s_{\text{trunc}} \in \mathbb{R}^{256 \times 768}$. To align with the acoustic sequence length, we apply one-dimensional average pooling over every two consecutive time steps in s_{trunc} . This yields the final semantic representation $O_s \in \mathbb{R}^{128 \times 768}$. During training, Whisper’s weights remain frozen to preserve high-level phoneme- and word-level consistency. This design forces the acoustic branch to align its features only when the semantic anchor O_s , indicates that spoken content is identical.

B. Coarse-to-Fine Acoustic Encoder

The goal of the acoustic branch is to isolate low-level codec fingerprints while filtering out irrelevant noise and background artifacts. We design a coarse-to-fine acoustic encoder: a fine-tuned Wav2Vec 2.0 XLS-R (0.3B) [30] first captures broad

codec-introduced anomalies, and a multi-decoder AudioMAE module then masks and reconstructs those embeddings to suppress noise and amplify residual, codec-specific fingerprints. Our multi-decoder AudioMAE is extended by a single decoder AudioMAE [31]. For simplicity, we refer to Wav2Vec 2.0 XLS-R (0.3B) as W2V2 in the following section. The view of the coarse-to-fine Acoustic Encoder is shown in Figure 1-(b).

In the first stage, a W2V2 model fine-tuned on the corresponding source-tracing task is employed to extract robust, coarse-grained representations associated with different codecs. The initiation of fine-tuning here aims to enable W2V2 to capture more neural codec characteristics. Given an input waveform x , the W2V2 generates initial embeddings e_{coarse} .

In the second stage, these embeddings are subsequently refined by a multi-decoder AudioMAE framework, which is inspired by the bottleneck architecture can purify some irrelevant information [32–34]. Within AudioMAE, the encoder’s output h is simultaneously passed to all decoders, each of which produces its own class-specific output $\{e_{\text{fine}}^i\}_{i=1}^N$, where N denotes the number of classification classes. Then, we use a multi-decoder scheme: since each class-specific autoencoder reconstructs inputs from its own class more accurately than others, the decoder with the best reconstruction indicates the corresponding class [35]. Leveraging this property, we employ a multi-head attention (MHA) mechanism to enable the model to focus on the reconstruction quality across all decoders. The informative representation h serves as the query (Q), while $e_{\text{fine}}^{\text{concat}}$, representing the concatenation of all reconstructed outputs from each decoder, serves as both the keys (K) and values (V). The MHA output O_a serves as the final acoustic feature, allowing a more accurate identification of the codec responsible for generating the input audio.

C. Semantic-Acoustic Fusion

The fusion module uses content-robust semantic embeddings as a stable anchor alongside acoustic features, which capture inherent codec-related patterns. Inspired by audio-visual synchronization [36, 37], a learnable cross-attention mechanism [38] then extracts semantic-acoustic mismatch features, which are inspired by audio-visual synchronization. Specifically, three Transformer modules learn these mismatches from different perspectives. Firstly, in the SA Transformer Encoder, semantic queries attend to acoustic keys/values to produce O_{SA} , forcing the semantic stream to highlight codec-induced distortions. Secondly, in the AS Transformer Encoder, acoustic queries attend to semantic keys/values to produce O_{AS} , downweighting any acoustic components aligned with the anchored content. Finally, these two conditioned maps are passed through a Fusion Transformer that uses O_{AS} as queries and O_{SA} as keys and values to yield O_{fusion} , which is then fed into the AASIST classifier.

D. Loss Function

The *SASTNet* integrates two loss functions: a classification loss L_{cls} from the AASIST backend and a reconstruction loss L_{recon} from AudioMAE. To balance these objectives, we adopt

the Automatic Weighted Loss (AWL) [39], which computes a weighted sum of the reconstruction and classification losses:

$$L_{\text{overall}} = \text{AWL}(L_{\text{cls}}, L_{\text{recon}}) \quad (1)$$

where two learnable weights, w_{cls} and w_{recon} , are automatically adjusted during training to optimize the contribution of each loss term. The classification loss L_{cls} is computed as the cross-entropy loss between the output scores of AASIST and the ground truth. The reconstruction loss L_{recon} for multi-decoder AudioMAE is defined as follows:

$$L_{\text{recon}} = L_{\text{MSE}_y} + \max(0, m + L_{\text{MSE}_y} - L_{\text{MSE}_{\text{other}}}) \quad (2)$$

where y denotes the input label, m denotes the *margin*, and L_{MSE_y} represents the Mean Squared Error between the masked region of the input e_{coarse} and the corresponding reconstructed region of output e_{fine}^y from *decoder_y*. $L_{\text{MSE}_{\text{other}}}$ refers to the average MSE between the masked region of e_{coarse} and corresponding reconstructed masked regions of other outputs e_{fine}^i from *decoder_i*, for $i \neq y$. Note that the input e_{coarse} is masked only during training. During inference, e_{coarse} is fed directly into the encoder without masking.

The goal is for the class-specific decoder to yield a lower reconstruction error compared to decoders not associated with the input class. This design ensures that each decoder is specialized in reconstructing inputs from its designated class. When the $L_{\text{MSE}_{\text{other}}}$ is lower than L_{MSE_y} , the second term in the loss becomes positive, increasing the overall reconstruction loss as a penalty. Conversely, if L_{MSE_y} is smaller than $L_{\text{MSE}_{\text{other}}}$ by at least m margin, the penalty term becomes zero, resulting in no additional loss. The margin $m = 0.1$ is set from observations that the MSE of the original single-decoder version of AudioMAE typically converges to 0.02–0.05. This setting enforces a minimum separation between the target decoder and the others, promoting class-specific specialization.

IV. SOURCE TRACING

A. Experimental Setup

We investigate source tracing in *CodecFake+* [21] dataset, which comprises two subsets: CoRS (speech resynthesized by pre-trained neural audio codec models) and CoSG (speech generated by codec-based speech generation models). For model training, we followed Chen et al. [22] and used taxonomy-guided balanced sampling to maximize data diversity under limited computational resources, selecting a subset of spoofed samples accordingly. The CoRS Train dataset contains 42,965 bona fide and 42,965 spoofed audio samples. Depending on the sampling strategy, different CoRS subsets are used. For example, when performing the AUX task, we use only the AUX subset of CoRS Train. For evaluation, we adopt the *CodecFake+* protocol, which includes three subsets: CoRS Test, CoSG Test (known codec), and CoSG Test (All).

Additionally, to assess the effects of unseen speech and silence on performance, we sampled subsets from CoRS: For training, we only use CoRS samples with utterance IDs ≤ 250 , referred to as the **CoRS Train (utt. ID ≤ 250) set**. For testing, we split into two subset: a **CoRS Test (Seen) set** with utterance IDs ≤ 250 and **CoRS Test (Unseen) set** > 250 .

TABLE I
F1 SCORE (%) \uparrow OF EVALUATION ON DIFFERENT SOURCE TRACING TASKS.

(a) Vector Quantization Classification			
Model	CoRS	CoSG (kn. codec)	CoSG (All)
Random	32.26	29.21	30.79
S-VQ [22]	97.39	42.94	35.26
SASTNet	98.94	46.94	48.17
(b) Auxiliary Objective Classification			
Model	CoRS	CoSG (kn. codec)	CoSG (All)
Random	28.78	30.94	31.04
S-AUX [22]	97.15	32.80	19.44
SASTNet	97.90	48.13	43.47
(c) Decoder Types Classification			
Model	CoRS	CoSG (kn. codec)	CoSG (All)
Random	38.97	36.89	34.37
S-DEC [22]	97.96	46.73	27.26
SASTNet	97.72	48.03	40.71

During training, the model sees only utterances from CoRS Train (utt. IDs ≤ 250), while CoRS Test (Unseen) evaluates its ability to generalize to new content. For a detailed discussion, please refer to Section V-A.

We adopt W2V2-AASIST [40]³ as backbone and train three baseline models on the CoRS Train set under different sampling strategies, referred to as S-VQ, S-AUX, and S-DEC. All input raw waveforms with a length of 82,200 samples at a sampling rate of 16,000 Hz, with RawBoost [41] augmentation. For AudioMAE a patch size of 16 is used, and during training, a masking ratio of 0.4 is applied, divided equally between time and spectral dimensions. No masking is applied during inference. We train models on an NVIDIA RTX 4090 GPU with a batch size of 12 for 40 epochs. The initial learning rate is 5×10^{-6} , and the weight decay is 1×10^{-4} .

B. Results

We present a comparison between the performance of SASTNet and baseline models across three source tracing tasks in Table I. Each row represents a distinct model, and corresponding F1 scores are derived from the model trained on the VQ, AUX, or DEC tasks. Each column denotes a different evaluation subset. Since this work focuses on improving single-task performance, we do not include multitask results for comparison with the previous work [22].

By examining F1-score changes from CoRS to CoSG (kn. codec), we see that SASTNet outperforms all baselines on the three tasks but still experiences roughly a 50% drop in F1-score. This shows that, while more robust than prior models, SASTNet remains affected by artifacts introduced during generative modeling. In CoSG (All), SASTNet’s performance in the AUX and DEC tasks drops by only 5%–8% compared to CoSG (kn. codec). Conversely, in the VQ task, SASTNet actually improves by 1.23% on CoSG (All) versus CoSG (kn. codec). Taken together, these results demonstrate that SASTNet generalizes well to unseen codecs and maintains strong source tracing performance across different generative modeling and codec scenarios.

³https://github.com/TakHemlata/SSL_Anti-spoofing

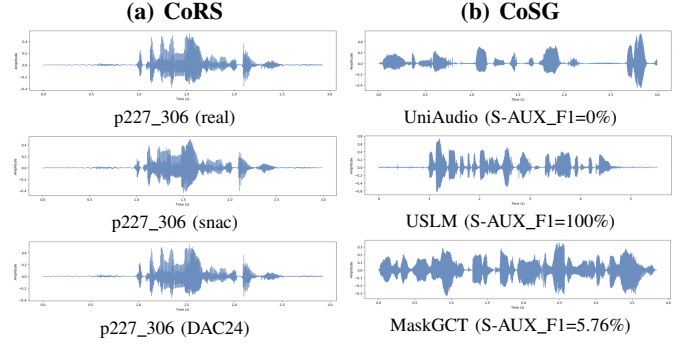


Fig. 2. The failure case study of codec-based deepfake source tracing.

TABLE II
F1 SCORES (%) \uparrow OF BASELINE MODELS TRAINED ON CoRS TRAIN (UTT. IDS ≤ 250) AND TESTED ON CoRS TEST (SEEN & UNSEEN)

Model	CoRS Test (Seen)			CoRS Test (Unseen)		
	original data	rm_sil	diff	original data	rm_sil	diff
S-VQ	93.62	69.04	-24.58	93.98	67.55	-26.43
S-AUX	93.70	69.45	-24.25	94.18	66.93	-27.25
S-DEC	92.05	68.21	-23.84	93.09	67.52	-25.57

V. ANALYSIS AND DISCUSSION

A. The Reason for Model Generalization Failures

Prior work [22] discussed how unseen codec types and generative modeling may affect model generalizability. In this section, we further identify two additional factors, silence and unseen content, that affect model generalization.

1) *Silence*: One challenge is silence variability: inconsistent pauses at utterance boundaries disrupt codec-signature alignment, causing models to focus on noise in silent segments rather than learning content-independent features. Figure 2(a) shows that CoRS contains silent intervals in both bona fide and resynthesized speech. This occurs because the CoRS samples in *CodecFake+* are derived from the untrimmed version of VCTK [42], which includes silence at the beginning and end of each utterance. Since silence is present in both bona fide and spoofed samples of different spoofing algorithms, it may go unnoticed during lab experiments but later leads to degradation when encountering samples with different silence characteristics in the real world. When the model is trained on CoRS with silence at the beginning and end of each utterance it can achieve perfect source-tracing F1 scores on samples with long silences as shown in Figure 2(b). However, the model suffers performance degradation when facing samples with short silences, e.g., from UniAudio [43] and MaskGCT [44]. We further analyzed the correlation between silence and performance by correlating the proportion of silent samples with F1 scores in CoSG ($\rho_{VQ} = 0.62$, $\rho_{AUX} = 0.60$, $\rho_{DEC} = 0.75$). These findings also appear in deepfake detection [45, 46] and highlight the need to model robustly against silence variability.

2) *Unseen Content or Speakers*: Unseen content and speakers are both notable challenges, which means that linguistic contexts or speaker information not present in the training set introduce phonetic and prosodic patterns outside the model’s learned distribution, thereby reducing performance. **Experiments**: We conducted a controlled experiment on CoRS set to isolate the effects of silence and variability in speech content

TABLE III

ABLATION ANALYSIS OF SILENCE AND UNSEEN SPEECH EFFECTS ON SASTNET. MODELS WERE TRAINED ON CoRS TRAIN (UTT. ID \leq 250) AND EVALUATED ON CoRS TEST (SEEN & UNSEEN) USING F1-SCORE.

Model	Front-end Modules		CoRS (Seen)		CoRS (Unseen)	
	W2V2 MAE	Whisper	orig.	rm_sil	orig.	rm_sil
S-AUX	✓		93.70	69.45	94.18	66.93
S-MAE [47]	✓		89.08	65.27	88.96	39.27
M-MAE		✓	93.42	67.73	92.66	55.30
Whisper + M-MAE		✓	85.01	71.45	84.16	60.29
Pretrained W2V2+ M-MAE	✓	✓	95.68	79.96	96.37	79.91
Tuned W2V2+ M-MAE	✓	✓	96.56	83.69	96.79	82.15
SASTNet (Pretrained W2V2)	✓	✓	95.10	86.54	95.29	86.22
SASTNet (Tuned W2V2)	✓	✓	98.74	89.77	98.51	89.76

on source-tracing performance. During training, the model was trained on the **CoRS Train (utt. IDs \leq 250) set**. For evaluation, we used **CoRS Test (Seen) set** and **CoRS Test (Unseen) set** to assess the model’s performance on both familiar and unfamiliar speech content. Additionally, we evaluated the impact of silence by comparing model performance on audio with and without silent segments, aiming to determine whether the model had over-fitted to silence.

Results: Table II presents the results of single-task baseline models evaluated on the CoRS Test (Unseen) set along with CoRS Test (Seen) set. Excluding silent segments from the input led to a substantial performance decline, with the single-task model’s F1 score decreasing by approximately 27%. This suggests that the presence of silence has a detrimental effect on the baseline models’ performance. When evaluated on seen speech, the F1 score is approximately 69%, whereas on unseen speech, it decreases to around 67%, indicating that the model’s effectiveness further diminishes when processing unseen speech content. This suggests that the CoSG dataset, which lacks silence and unseen speech components relative to CoRS, poses a challenge for models to generalize to CoSG.

In summary, in Sections IV-B and V-A, we show how four variables, namely generative modeling process, unseen codec types, silence diversity, and unseen speech, combine to limit model generalization and tracing fidelity.

B. The Impact of Semantic and Acoustic Encoder

To gain deeper insights into how the semantic and acoustic encoders contribute to generalization under the effects of silence, unseen speech, codec type, and generative modeling, we conducted an ablation study on the AUX task and present the results in Tables III and IV. We compare models using different semantic and acoustic encoders by evaluating various front-end configurations as follows:

- **MAE-only:** This configuration uses mel-spectrograms as input to the MAE module. We further divide this setting into S-MAE (Single-Decoder MAE) and M-MAE (Multi-Decoder MAE) based on the number of decoders used.
- **Whisper + MAE:** The MAE processes mel-spectrograms to extract acoustic features, which are then fused with semantic features extracted from Whisper.
- **W2V2 + MAE:** Here, we replace the MAE’s mel-spectrogram input with W2V2 SSL features. We consider

TABLE IV

ABLATION STUDY OF CODEC TYPES AND GENERATIVE MODELING EFFECTS ON SASTNET. THE MODEL WAS TRAINED ON THE CoRS TRAIN SET USING AUX BALANCED SAMPLING AND EVALUATED ON THE STANDARD *CodecFake+* EVALUATION SET.

Model	Front-end Modules			F1 (%) \uparrow		
	W2V2	MAE	Whisper	CoRS	CoSG (kn.)	CoSG (All)
Random				28.78	30.94	31.04
S-AUX	✓			97.15	32.80	19.44
S-MAE [47]		✓		96.67	42.21	35.59
M-MAE		✓		97.10	38.41	36.64
Whisper + M-MAE		✓	✓	95.98	41.74	36.94
Pretrained W2V2+ M-MAE	✓	✓		98.62	43.29	36.89
Tuned W2V2+ M-MAE	✓	✓		95.81	46.18	41.91
SASTNet (Pretrained W2V2)	✓	✓	✓	94.80	47.28	40.62
SASTNet (Tuned W2V2)	✓	✓	✓	97.90	48.13	43.47

two variants: Pretrained W2V2 + M-MAE, where W2V2 is initialized with pretrained weights; and Tuned W2V2 + M-MAE, where W2V2 is initialized with the S-AUX task-specific weights.

- **Our proposed SASTNet** has two variants: SASTNet (Pretrained W2V2), where W2V2 is initialized with pretrained weights; and SASTNet (Tuned W2V2), where W2V2 is initialized with the S-AUX weights.

1) *Effects of Silence and Unseen Speech.*: In Table III, compared to S-AUX, using MAE alone as the front-end resulted in a significant decline in F1 score due to the removal of silence, with S-MAE and M-MAE showing drops of 49.69 and 37.36, respectively. Notably, M-MAE demonstrated greater robustness to silence removal than S-MAE. The Whisper+MAE exhibited a smaller F1 drop of 23.87, indicating higher robustness compared to both S-AUX and the MAE-only settings. Similarly, W2V2+MAE showed improved robustness relative to MAE-only, with Pretrained W2V2+M-MAE and Tuned W2V2+M-MAE resulting in drops of 16.46 and 14.64, respectively. This suggests that initializing W2V2 with S-AUX improves robustness to silence. Finally, SASTNet (Pretrained W2V2) and SASTNet (Tuned W2V2) exhibited the smallest performance degradation, with drops of 9.07 and 8.75, respectively—approximately half the drop observed with W2V2+MAE.

For unseen speech content without silence, F1 scores improve across models, from 39.27 for S-MAE to 89.76 for SASTNet (Tuned W2V2), indicating that multi-decoder reconstruction, Fine-tuned W2V2, and Whisper-based features each play a crucial role in enhancing robustness under silence removal and unseen speech conditions. Moreover, mel-spectrogram inputs exhibit pronounced sensitivity to speech content: when the input of MAE is mel-spectrograms, the F1 gap between seen and unseen speech is considerable. In contrast, substituting mel-spectrograms with SSL features markedly reduces this discrepancy, indicating that SSL features are more invariant to content than mel-spectrograms.

2) *Effects of Codec Type and Generative Modeling.*: In Table IV, the performance of S-AUX on CoSG is notably poor, even falling below the F1 score of the random baseline. This

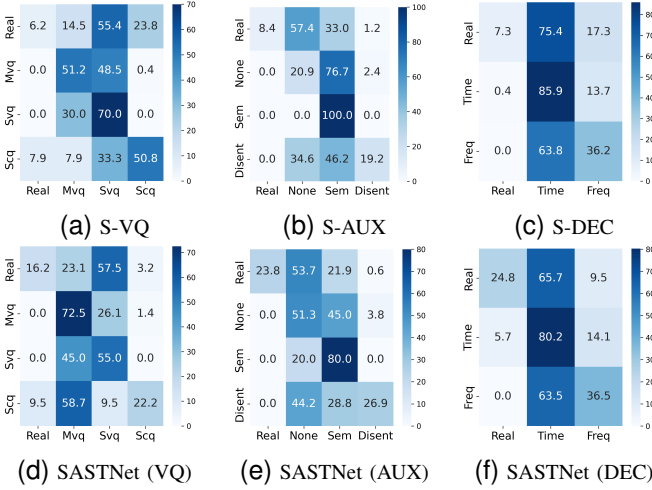


Fig. 3. Confusion matrices of baseline and SASTNet on CoSG (All) evaluation set, row-normalized by true labels, with predictions on the horizontal axis and true labels on the vertical axis.

indicates that *S-AUX* fails to perform source tracing on CoSG data. In contrast, using MAE as the front-end yields better results. Specifically, on CoSG (All), *S-MAE* achieves a 35.59 F1 score, while *M-MAE* reaches 36.64. Across the 17 CoSG models, *M-MAE* consistently outperforms *S-MAE* in three unseen codecs and in 9 out of 14 seen codecs. These results suggest that multi-decoder reconstruction improves robustness when handling unseen codecs and generative modeling.

Moreover, (*Whisper+MAE*) integrating *Whisper* with *MAE* leads to a slight performance improvement over *MAE* alone, achieving F1 scores of 41.74 on CoSG (kn.) and 36.94 on CoSG (All). Similarly, incorporating *W2V2* with *MAE* (*W2V2+MAE*) also enhances performance. Specifically, *Pretrained W2V2+M-MAE* yields F1 scores of 43.29 on CoSG (kn.) and 36.89 on CoSG (All), while *Tuned W2V2+M-MAE* achieves 46.18 and 41.91, respectively. These findings indicate that SSL-based features are more robust than traditional mel-spectrograms in handling unseen codecs and generative modeling variations. The superior performance of the *Tuned W2V2* could be attributed to its prior knowledge to codec-related data, whereas the *Pretrained W2V2* was only pre-trained on bona fide samples. As a result, the *Tuned W2V2* is better equipped to capture codec-specific artifacts.

Finally, *SASTNet* (*Pretrained W2V2*) achieves F1 scores of 47.28 on CoSG (kn.) and 40.62 on CoSG (All), while *SASTNet* (*Tuned W2V2*) achieves 48.13 and 43.47, respectively. These results represent a 24% F1 improvement over *W2V2+MAE*, highlighting the effectiveness of fusing refined SSL and *Whisper* features. This fusion enables the model to better focus on codec-related artifacts with semantic anchors, thereby enhancing performance on CoSG data.

C. Discussion

To better understanding results, we compared the confusion matrices of *SASTNet* and the baseline on the CoSG (All) dataset, with the results shown in Fig. 3. Notably, *SASTNet* demonstrates significantly improved accuracy in detecting unseen real speech across the VQ, AUX, and DEC source tracing

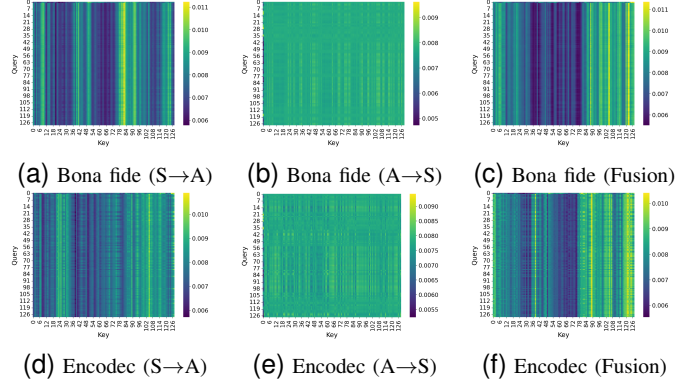


Fig. 4. The Semantic-Acoustic Interaction Attention Maps for Case Study.

tasks. Specifically, on CoSG (All), the accuracy for unseen real speech improves by $2.63\times$, $2.7\times$, and $3.36\times$ compared to the baseline, respectively. Moreover, *SASTNet* shows improved performance in identifying MVQ, None, Disentanglement, and Frequency-domain codecs, suggesting that it generalizes better across various codec attributes. This can be attributed to its joint modeling of semantic and acoustic features, which allows the model to compare bona fide and codec-resynthesized versions of the same utterance. By leveraging consistent semantic content with contrast acoustic cues, *SASTNet* is better equipped to capture subtle artifacts left from codecs, thereby enhancing its detection performance.

In Figure 4, we visualize three cross-attention maps from the last layer of each transformer encoder for a randomly chosen bona fide Encoder synthesis pair. Figures (a) and (d) show opposite highlighted regions, indicating that the SA Transformer focuses on semantic-acoustic mismatches information (codec distortions). For resynthesized audio, this produces a wider, more dispersed bright band. Figures (b) and (e) remain flat and uniform for bona fide speech, and only exhibit slight brightness at distorted frames for spoofed audio. This shows that the AS module preserves content-aligned regions while suppressing interference. Figures (c) and (f) combine both patterns in spoofed audio, with highlights at content-aligned and distorted information. This implies that the Fusion module merges both patterns so the classifier can focus on codec-specific fingerprints, regardless of content, thus improving detection precision and generalization.

VI. CONCLUSION

To sum up, we have shown that models trained on neural codec re-synthesised speech often overfit to silence regions and fail to generalize across diverse silence conditions or unseen content. To overcome these limitations, we introduced *SASTNet*, which combines a semantic encoder with a coarse-to-fine acoustic encoder to simultaneously preserve linguistic content and extract detailed codec fingerprints. Empirical results confirm that *SASTNet* substantially improves performance in codec-based deepfake source tracing, demonstrating its effectiveness and robustness.

VII. ACKNOWLEDGEMENTS

This work was partially supported by the National Science and Technology Council, Taiwan (Grant no. NSTC 112-2634-F-002-005, Advanced Technologies for Designing Trustable AI Services). We also thank the National Center for High-performance Computing (NCHC) of National Applied Research Laboratories (NARLabs) in Taiwan for providing computational and storage resources.

VIII. BROADER IMPACTS

In recent years, general-purpose audio language models have become foundational for speech understanding [48–50]. However, even state-of-the-art systems like GPT-4o still steer clear of deepfake detection because of safety restrictions [51], underscoring their blind spots when it comes to synthetic audio threats. By contrast, SASTNet provides a targeted, interpretable framework specifically designed to trace codec-based speech synthesis. We hope our approach sheds light on this critical problem and spurs further exploration in the field.

REFERENCES

- [1] M. Li, Y. Ahmadiadli, and X.-P. Zhang, “A survey on speech deepfake detection,” *ACM Comput. Surv.*, vol. 57, no. 7, 2025.
- [2] H. Wu, J. Kang, L. Meng, H. Meng, and H.-y. Lee, “The defender’s perspective on automatic speaker verification: An overview,” *arXiv preprint arXiv:2305.12804*, 2023.
- [3] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, C. Hanilçi, M. Sahidullah, and A. Sizov, “Asvspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge,” in *Proc. Interspeech*, 2015.
- [4] T. Kinnunen, M. Sahidullah, H. Delgado, M. Todisco, N. Evans, J. Yamagishi, and K. A. Lee, “The asvspoof 2017 challenge: Assessing the limits of replay spoofing attack detection,” in *Proc. Interspeech*, 2017.
- [5] M. Todisco, X. Wang, V. Vestman, M. Sahidullah, H. Delgado, A. Nautsch, J. Yamagishi, N. Evans, T. H. Kinnunen, and K. A. Lee, “ASVspoof 2019: future horizons in spoofed and fake audio detection,” in *Proc. Interspeech*, 2019.
- [6] X. Liu, X. Wang, M. Sahidullah, J. Patino, H. Delgado, T. Kinnunen, M. Todisco, J. Yamagishi, N. Evans, A. Nautsch, and K. A. Lee, “ASVspoof 2021: Towards spoofed and deepfake speech detection in the wild,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 31, 2023.
- [7] X. Wang, H. Delgado, H. Tak, J.-w. Jung, H.-j. Shim, M. Todisco *et al.*, “ASVspoof 5: Crowdsourced speech data, deepfakes, and adversarial attacks at scale,” in *Proc. ASVspoof Workshop*, 2024.
- [8] J. Yi, R. Fu, J. Tao, S. Nie, H. Ma, C. Wang, T. Wang *et al.*, “Add 2022: the first audio deep synthesis detection challenge,” in *Proc. ICASSP*, 2022.
- [9] J. Yi, C. Y. Zhang, J. Tao, C. Wang, X. Yan, Y. Ren, H. Gu, and J. Zhou, “Add 2023: Towards audio deepfake detection and analysis in the wild,” *arXiv preprint arXiv:2408.04967*, 2024.
- [10] C. Borrelli, P. Bestagini, F. Antonacci, A. Sarti, and S. Tubaro, “Synthetic speech detection through short-term and long-term prediction traces,” *EURASIP Journal on Information Security*, vol. 2021, 2021.
- [11] X. Yan, J. Yi, J. Tao, C. Wang, H. Ma, T. Wang, S. Wang, and R. Fu, “An initial investigation for detecting vocoder fingerprints of fake audio,” *Proceedings of the 1st International Workshop on Deepfake Detection for Audio Multimedia*, 2022.
- [12] C. Y. Zhang, J. Yi, J. Tao, C. Wang, and X. Yan, “Distinguishing neural speech synthesis models through fingerprints in speech waveforms,” in *China National Conference on Chinese Computational Linguistics*, 2023.
- [13] T. Zhu, X. Wang, X. Qin, and M. Li, “Source tracing: detecting voice spoofing,” in *Proc. APSIPA ASC*, 2022.
- [14] N. Klein, T. Chen, H. Tak, R. Casal, and E. Khoury, “Source tracing of audio deepfake systems,” in *Proc. Interspeech*, 2024.
- [15] S. Arora, K.-W. Chang, C.-M. Chien, Y. Peng, H. Wu, Y. Adi, E. Dupoux, H.-Y. Lee, K. Livescu, and S. Watanabe, “On the landscape of spoken language models: A comprehensive survey,” *arXiv preprint arXiv:2504.08528*, 2025.
- [16] Y. Xie, X. Wang, Z. Wang, R. Fu, Z. Wen, S. Cao, L. Ma, C. Li, H. Cheng, and L. Ye, “Neural codec source tracing: Toward comprehensive attribution in open-set condition,” *arXiv preprint arXiv:2501.06514*, 2025.
- [17] J. Mishra, M. Chhibber, H.-j. Shim, and T. H. Kinnunen, “Towards explainable spoofed speech attribution and detection: a probabilistic approach for characterizing speech synthesizer components,” *arXiv preprint arXiv:2502.04049*, 2025.
- [18] V. Negroni, D. Salvi, P. Bestagini, and S. Tubaro, “Source verification for speech deepfakes,” *arXiv preprint arXiv:2505.14188*, 2025.
- [19] Y. Xiao and R. K. Das, “Listen, analyze, and adapt to learn new attacks: An exemplar-free class incremental learning method for audio deepfake source tracing,” *arXiv preprint arXiv:2505.14601*, 2025.
- [20] H. Wu, Y. Tseng, and H. yi Lee, “CodecFake: Enhancing anti-spoofing models against deepfake audios from codec-based speech synthesis systems,” in *Proc. Interspeech*, 2024.
- [21] X. Chen, J. Du, H. Wu, L. Zhang, I. Lin, I. Chiu, W. Ren, Y. Tseng, Y. Tsao, J.-S. R. Jang, and H.-y. Lee, “CodecFake+: A large-scale neural audio codec-based deepfake speech dataset,” *arXiv preprint arXiv:2501.08238*, 2025.
- [22] X. Chen, I. Lin, L. Zhang, J. Du, H. Wu, H.-y. Lee, J.-S. R. Jang *et al.*, “Codec-based deepfake source tracing via neural audio codec taxonomy,” *arXiv preprint arXiv:2505.12994*, 2025.
- [23] Y. Guo, Z. Li, H. Wang, B. Li, C. Shao, H. Zhang, C. Du, X. Chen, S. Liu, and K. Yu, “Recent advances in discrete speech tokens: A review,” *arXiv preprint arXiv:2502.06490*, 2025.
- [24] H. Wu, X. Chen, Y.-C. Lin, K.-w. Chang, H.-L. Chung, A. H. Liu, and H.-y. Lee, “Towards audio language modeling-an overview,” *arXiv preprint*

- arXiv:2402.13236*, 2024.
- [25] H. Wu, H.-L. Chung, Y.-C. Lin, Y.-K. Wu, X. Chen, Y.-C. Pai *et al.*, “Codec-SUPERB: An in-depth analysis of sound codec models,” in *Findings Assoc. Comput. Linguist.*, 2024.
 - [26] H. Wu, X. Chen, Y.-C. Lin, K. Chang, J. Du, K.-H. Lu *et al.*, “Codec-SUPERB@ SLT 2024: A lightweight benchmark for neural audio codec models,” in *Proc. IEEE Spoken Lang. Technol. Workshop*, 2024.
 - [27] P. Kawa, M. Plata, M. Czuba, P. Szymański, and P. Syga, “Improved deepfake detection using whisper features,” in *Interspeech 2023*, 2023.
 - [28] J.-w. Jung, H.-S. Heo, H. Tak, H.-j. Shim, J. S. Chung, B.-J. Lee, H.-J. Yu, and N. Evans, “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)*, 2022.
 - [29] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, “Robust speech recognition via large-scale weak supervision,” 2022.
 - [30] A. Babu, C. Wang, A. Tjandra, K. Lakhota, Q. Xu, N. Goyal, K. Singh, P. Von Platen, Y. Saraf, J. Pino *et al.*, “Xls-r: Self-supervised cross-lingual speech representation learning at scale,” *arXiv preprint arXiv:2111.09296*, 2021.
 - [31] P.-Y. Huang, H. Xu, J. Li, A. Baevski, M. Auli, W. Galuba, F. Metze, and C. Feichtenhofer, “Masked autoencoders that listen,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 28 708–28 720, 2022.
 - [32] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, “Mae: Masked autoencoders are scalable vision learners,” *arXiv preprint arXiv:2111.06377*, 2021.
 - [33] H. Wu, P.-C. Hsu, J. Gao, S. Zhang, S. Huang, J. Kang, Z. Wu, H. Meng, and H.-Y. Lee, “Adversarial sample detection for speaker verification by neural vocoders,” in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.
 - [34] X. Chen, J. Du, H. Wu, J.-S. R. Jang, and H. yi Lee, “Neural codec-based adversarial sample detection for speaker verification,” in *Interspeech 2024*, 2024.
 - [35] R. Du, J. Yao, Q. Kong, and Y. Cao, “Towards out-of-distribution detection in vocoder recognition via latent feature reconstruction,” *arXiv preprint arXiv:2406.02233*, 2024.
 - [36] X. Chen, H. Wu, C.-C. Wang, H.-Y. Lee, and J.-S. R. Jang, “Multimodal transformer distillation for audio-visual synchronization,” in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024.
 - [37] X. Chen, H. Wu, H. Meng, H.-y. Lee, and J.-S. R. Jang, “Push-pull: Characterizing the adversarial robustness for audio-visual active speaker detection,” in *2022 IEEE Spoken Language Technology Workshop (SLT)*, 2023.
 - [38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.
 - [39] L. Liebel and M. Körner, “Auxiliary tasks in multi-task learning,” *arXiv preprint arXiv:1805.06334*, 2018.
 - [40] H. Tak, M. Todisco, X. Wang, J.-w. Jung, J. Yamagishi, and N. Evans, “Automatic speaker verification spoofing and deepfake detection using wav2vec 2.0 and data augmentation,” in *Proc. Odyssey Speaker Lang. Recognit. Workshop*, 2022.
 - [41] H. Tak, M. Kamble, J. Patino, M. Todisco, and N. Evans, “Rawboost: A raw data boosting and augmentation method applied to automatic speaker verification anti-spoofing,” in *Proc. ICASSP*, 2022.
 - [42] J. Yamagishi, C. Veaux, K. MacDonald *et al.*, “Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit (version 0.92),” *Univ. of Edinburgh, The Centre for Speech Technology Research (CSTR)*, 2019.
 - [43] D. Yang, J. Tian, X. Tan, R. Huang, S. Liu, H. Guo, X. Chang, J. Shi, S. Zhao, J. Bian, Z. Zhao, X. Wu, and H. M. Meng, “UniAudio: Towards universal audio generation with large language models,” in *Proceedings of the 41st International Conference on Machine Learning*, 2024.
 - [44] Y. Wang, H. Zhan, L. Liu, R. Zeng, H. Guo, J. Zheng *et al.*, “MaskGCT: Zero-shot text-to-speech with masked generative codec transformer,” in *Proc. ICLR*, 2025.
 - [45] Y. Zhang, Z. Li, J. Lu, H. Hua, W. Wang, and P. Zhang, “The impact of silence on speech anti-spoofing,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 3374–3389, 2023.
 - [46] Y. Zhang¹², W. Wang¹², and P. Zhang¹², “The effect of silence and dual-band fusion in anti-spoofing system,” in *Proc. Interspeech*, 2021, pp. 4279–4283.
 - [47] X. Wang, R. Fu, Z. Wen, Z. Wang, Y. Xie, Y. Liu, J. Tao, X. Liu, Y. Li, X. Qi *et al.*, “Genuine-focused learning using mask autoencoder for generalized fake audio detection,” *arXiv preprint arXiv:2406.03247*, 2024.
 - [48] C.-y. Huang, K.-H. Lu, S.-H. Wang, C.-Y. Hsiao, C.-Y. Kuan, H. Wu, S. Arora, K.-W. Chang, J. Shi, Y. Peng *et al.*, “Dynamic-superb: Towards a dynamic, collaborative, and comprehensive instruction-tuning benchmark for speech,” in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 12 136–12 140.
 - [49] C.-Y. Huang *et al.*, “Dynamic-SUPERB phase-2: A collaboratively expanding benchmark for measuring the capabilities of spoken language models with 180 tasks,” in *Proc. ICLR*, 2025.
 - [50] C.-K. Yang, N. S. Ho, and H.-y. Lee, “Towards holistic evaluation of large audio-language models: A comprehensive survey,” *arXiv preprint arXiv:2505.15957*, 2025.
 - [51] Y.-X. Lin, C.-K. Yang, W.-C. Chen, C.-A. Li, C.-Y. Huang, X. Chen, and H.-y. Lee, “A preliminary exploration with gpt-4o voice mode,” *arXiv preprint arXiv:2502.09940*, 2025.