CLEANCODEC: EFFICIENT AND ROBUST SPEECH TOKENIZATION VIA PERCEPTUALLY GUIDED ENCODING

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural audio codecs are a key component of speech processing pipelines, compressing audio into discrete tokens for downstream modeling. However, existing codecs struggle to balance reconstruction quality with token efficiency, often encoding perceptually irrelevant information such as background noise and recording artifacts at the expense of linguistically and acoustically meaningful content. We reframe audio tokenization as a selective information bottleneck problem and propose CleanCodec, a neural audio codec utilizing a *denoising* training objective, where the codec receives stochastically degraded speech as input but is trained to reconstruct the corresponding clean signal. CleanCodec learns to encode only perceptually important features and achieves state-of-the-art tokenization efficiency at just 12.5 t/s, improving speaker similarity by 0.21 and reducing word error rate by up to 16.3% compared to existing codecs. Evaluations on downstream text-to-speech and voice conversion tasks further demonstrate improved performance and up to 17x faster inference, highlighting significant efficiency gains due to reduced token rate. [1]

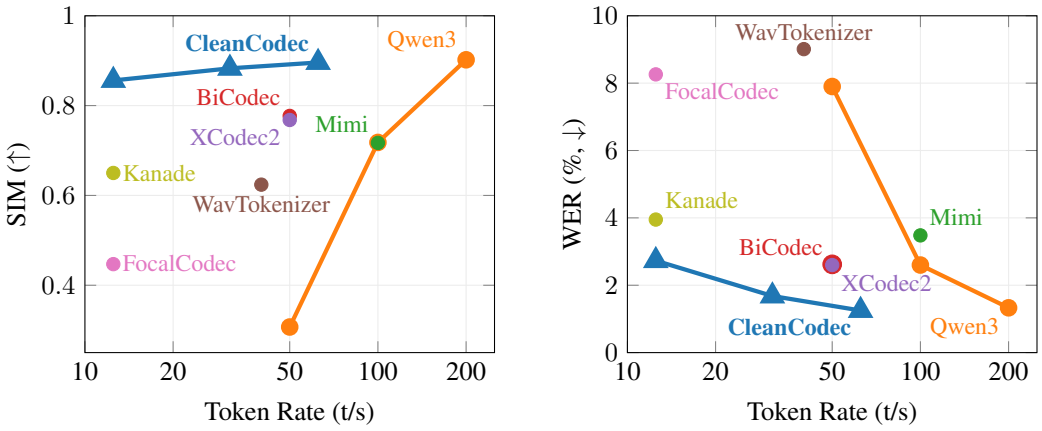

Figure 1: Left: SIM vs. Token rate. Right: WER vs. Token rate. CleanCodec substantially improves over existing codecs with similar token rate on LibriTTS test-clean (Zen et al., 2019).

1 INTRODUCTION

The rise of large language model (LLM)-based audio generation, latent diffusion-based text-to-speech (TTS), and multimodal speech integration has driven an urgent need for efficient discrete acoustic representations. A key component of these pipelines is the neural audio codec, which compresses speech into token sequences that downstream models can process. To achieve high reconstruction quality, these tokens must preserve both semantic information (the linguistic content of what is being said) and acoustic information (speaker timbre, prosody, and other non-linguistic properties).

A key gap between the audio and text modalities is tokenization efficiency. Unlike raw text, audio waveforms of speech contain non-linguistic information such as speaker identity, prosody, and other

[1] Listening samples, models and code will be released on GitHub.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

acoustic information. As a result, audio codecs require significantly higher token rates than text tokenizers—often hundreds of tokens per second of audio. This presents significant challenges for LLM-based audio approaches; at 100 t/s, encoding a one-minute clip already requires 6,000 tokens. High token rates increase costs and make it challenging to capture long-term dependencies. Encoding this information using as few tokens as possible remains a primary focus to improve the efficiency of downstream audio tasks.

In this work, we reframe speech tokenization as a *selective information bottleneck* problem. We argue that rather than treating audio compression as a task of maximally preserving all signal information, the ideal audio codec should only encode salient information and discard perceptually insignificant features such as background noise and recording artifacts. To operationalize this principle, we propose a denoising codec objective in which the encoder receives a stochastically degraded utterance while the model reconstructs the corresponding clean utterance. Unlike inference-time enhancement preprocessing, this objective shapes what information passes through the discrete bottleneck during training. We then use this objective to train CleanCodec, using auxiliary semantic and speaker-conditioning losses to preserve linguistic content and timbre. At just 12.5 tokens per second, CleanCodec achieves SOTA performance among codecs operating at comparable token rates, advancing the Pareto frontier in both SIM and WER (See Figure 1). These gains also transfer into downstream tasks, where CleanCodec improves speaker similarity ($0.45 \rightarrow 0.56$) and word error rate ($5.6\% \rightarrow 3.9\%$) scores compared to the strongest 12.5 t/s baseline in Seed-TTS-Eval (Anastassiou et al., 2024).

Our contributions are as follows: i) We propose a denoising training objective for neural audio codecs: the model receives noisy speech as input and must reconstruct the clean audio, prioritizing the preservation of only salient information. ii) We apply this objective in CleanCodec, a low-token-rate denoising audio codec with auxiliary semantic and speaker-conditioning losses. According to benchmarks, CleanCodec achieves superior reconstruction for a given information budget compared to existing audio codecs, demonstrating superior encoding efficiency. iii) We demonstrate how the resulting representations improve reconstruction, robustness, and downstream tasks such as voice conversion and TTS.

2 Related Work

Multi-codebook designs. Existing approaches to speech tokenization broadly fall into two families. Semantic tokenizers use SSL models like HuBERT (Hsu et al., 2021) and wav2vec2 (Baevski et al., 2020) to map audio to linguistically meaningful representations via k-means clustering, capturing phonetic structure well but discarding prosodic and acoustic detail (Nguyen et al., 2023). Acoustic codecs (Défossez et al., 2023; Chen et al., 2025; Zhang et al., 2024; Li et al., 2025) instead train encoder-decoder networks (van den Oord et al., 2017) with reconstruction losses and vector quantization (VQ) (Gray, 1984) to discretize audio into compact code vectors. While acoustic codecs achieve high-fidelity reconstruction, they typically rely on multi-codebook designs such as residual vector quantization (RVQ) (Chen et al., 2010) — for instance, DAC (Kumar et al., 2023b) employs 9 RVQ layers — adding substantial complexity to downstream modeling and token rates that far exceed text-focused tokenizers.

Single-codebook designs. Single-codebook designs have emerged as a promising alternative, streamlining both training and inference while enabling more seamless integration with text-based language models (Parker et al., 2025). WavTokenizer (Ji et al., 2025) achieves 40 tokens per second using k-means clustering and random awakening strategies, and Single-Codec (Li et al., 2024) leverages a specialized BiLSTM (Hochreiter & Schmidhuber, 1997) architecture to preserve reconstruction quality with a single quantizer. More recently, FocalCodec (Libera et al., 2025) introduces a focal modulation-based compressor-quantizer-decompressor architecture operating at ultra-low token rates (12.5 - 50 t/s) with a single binary codebook. However, our experiments show that these single-codebook approaches continue to struggle to balance compression with both reconstruction quality and effective semantic representation, particularly at low token rates.

Disentangled codec designs. A parallel line of work has pursued disentangled codecs (Ju et al., 2024; Pan et al., 2024; Ren et al., 2024), which separate audio into time-invariant global representations and local audio tokens. Kanade (Huang et al., 2026) uses SSL features as encoder inputs alongside a narrow information bottleneck to achieve clean, unsupervised disentanglement. By operating on SSL features, Kanade inherits a structured latent space in which content and speaker

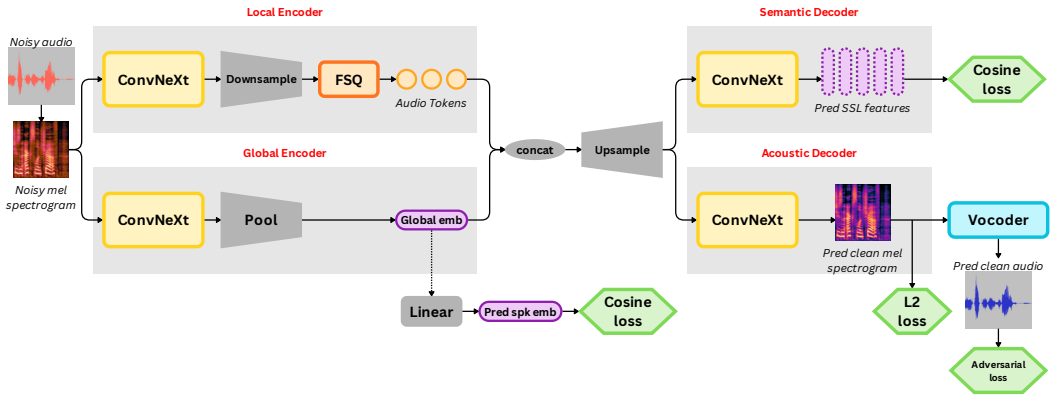

Figure 2: Architecture of CleanCodec, along with loss sources used during training.

information are easily separable, yielding strong lexical availability and prosody preservation with a single token stream. Nevertheless, using SSL features as inputs limits generalization to out-of-distribution and unseen acoustic domains, and the use of recurrent or attention-based sequence modeling constrains the codec to the input lengths seen during training.

Our novelty. Speech enhancement is commonly applied as a separate preprocessing step before encoding (Nakata et al., 2026). In contrast, our contribution is a codec-level denoising objective: degradations are applied during training, and the discrete representation of the degraded input is supervised toward the corresponding clean target. This directly shapes the information retained by the codec bottleneck rather than merely cleaning its input at inference time.

3 Methodology

3.1 Architecture

Our architecture follows commonly used designs and is illustrated in Figure 2. CleanCodec comprises (1) a disentangled VQ-VAE-style autoencoder operating on log-mel spectrograms and (2) a Vocos vocoder (Siuzdak, 2024) to convert mel spectrograms into raw audio. When encoding audio, CleanCodec converts the audio waveform to a mel spectrogram, encodes the spectrogram into discrete tokens, reconstructs the original mel spectrogram, then converts the mel spectrogram back into a waveform. Notably, this avoids relying on SSL features as input, which can lead to loss of non-linguistic information and a lack of robustness. We argue that mel features are also superior to using raw audio waveforms as input, as used by HH-Codec (Xue et al., 2025) and VibeVoice (Peng et al., 2025), because mel spectrograms align more closely with human perception of audio. Converting the input audio to a mel spectrogram reduces the temporal resolution from 24 kHz to 62.5 Hz.

Local Encoder. We adopt a fully convolutional architecture, using 1D ConvNeXt (Liu et al., 2022) blocks as proposed in Vocos (Siuzdak, 2024). After passing through the local encoder, encoder outputs are further downsampled by a factor of 5 to reach the desired token rate of 12.5 t/s. Then, we use finite scalar quantization (FSQ) (Mentzer et al., 2024) to quantize encoder outputs. FSQ is single-codebook, parameter-free, and avoids codebook divergence caused by traditional VQ. We use codebook levels of [8, 8, 8, 8, 8] for a total of 32,768 possible codes.

Global Encoder. Our global encoder architecture is inspired by Kanade (Huang et al., 2026), using modified ConvNeXt blocks from NeXt-TDNN (Heo et al., 2024). After passing through the global encoder, we apply attentive statistics pooling (Okabe et al., 2018) to obtain a single global embedding for the entire utterance. We use a global embedding size of 256, which is concatenated to the audio tokens during the decoding phase.

Semantic-Acoustic Dual Decoder. Audio codecs are frequently trained to encode both acoustic and semantic features to best preserve linguistic and non-linguistic information. We achieve this by employing a dual-decoder architecture. After receiving audio tokens augmented with the global embedding, we upsample to recover the original temporal resolution of 62.5 Hz. Then, the path of computation splits into two branches. The *acoustic decoder* reconstructs the original mel spectrogram,

while the *semantic decoder* reconstructs SSL features produced by WavLM-large (Chen et al., 2022), a self-supervised audio model with rich semantic features. This ensures that our token representation contains both semantic and acoustic information, leading to high-quality reconstruction. As in the local encoder, we use 1D ConvNeXt blocks for both decoders.

Vocoder. Finally, the reconstructed mel spectrogram is passed through a Vocos-based vocoder (Siuzdak, 2024) to convert it to an audio waveform. The vocoder is an important aspect of the codec, as it directly influences reconstruction quality. Therefore, we train our own Vocos model with increased depth and width for better reconstruction.

3.2 Training Framework

Motivation. As token rate decreases, it becomes increasingly challenging to accurately reconstruct audio. This is because audio is composed of a mixture of *perceptually important* information, such as linguistic content and speaker timbre, as well as *perceptually unimportant* information, such as background noise, reverberation, or microphone quality. Existing audio codecs attempt to preserve all information in the original audio, typically by reconstructing the mel spectrogram or raw waveform. However, we argue that some aspects of audio are more important than others. The ideal audio codec should ignore all perceptually unimportant information and allocate its limited bitrate to encoding only perceptually important components of audio, since this will maximize the perceived quality of the reconstructed audio. With this principle, we introduce a denoising objective that explicitly determines which information the codec learns to retain.

Speech degradation pipeline. In addition to training with a standard audio reconstruction task, we jointly use a speech enhancement training objective. During training, we corrupt speech utterances with a variety of audio degradations and supervise the model to reconstruct the original clean signal. This causes our model to ignore and remove perceptually unimportant information added by these degradations, allowing it to prioritize encoding important linguistic and auditory information. We use a degradation pipeline based on Nakata et al. (2026) that leaves linguistic content and speaker identity unchanged, with each degradation applied with a probability of 0.5:

- Reverberation: We sample an RT60 value and rectangular room dimensions from $\mathcal{U}(0.1, 2.0)$ seconds and $\mathcal{U}(2, 20)$ m, respectively. Then, we use pyroomacoustics (Scheibler et al., 2018) to simulate room impulse responses (RIRs). We filter any RIRs longer than 50,000 samples to avoid excessive reverberation.
- Background noise: We randomly sample a noise recording from a dataset pool of AudioSet (Gemmeke et al., 2017), FSD50K (Fonseca et al., 2022), and WHAM! (Wichern et al., 2019). We add this to the clean utterance with an SNR sampled from $\mathcal{U}(15, 30)$ dB.
- Low-pass filter: We use a biquad low-pass filter with a cutoff frequency ranging from 2–8 kHz.
- Resampling: The utterance is resampled to one of $\{16, 22.05\}$ kHz before being resampled back to 24 kHz.
- Codec: We apply MP3 compression with a random bitrate between 32–245 kbps.

Global Conditioning. Conditioning model outputs on SSL features is a frequently employed technique to better preserve linguistic information. However, we believe that speaker timbre is also important to preserve. As an auxiliary component, we condition the global encoder using a pretrained speaker-verification (SV) model to better preserve speaker timbre. Specifically, we use TitaNet-L (Koluguri et al., 2022), a pretrained speaker recognition model trained on over 100,000 speakers. We apply a learned linear projection from our global embedding to TitaNet's embedding space and train the encoder to maximize its cosine similarity with the pretrained embeddings. This speaker-conditioning loss is a secondary novel contribution that improves speaker reconstruction.

Training objectives. Our notations in this work are listed in Table 1. Importantly, the encoder receives $\tilde{m}$ as input, whereas the acoustic decoder is supervised using the clean target m. The autoencoder uses three loss sources: i) L2 loss between the original mel spectrogram and the output produced by the acoustic decoder, denoted $\mathcal{L}_{\text{mel}}$:

$$\mathcal{L}_{\text{mel}} = ||m - \hat{m}||_2.$$

ii) Cosine loss between the SSL features from WavLM-large and the semantic decoder output, denoted $\mathcal{L}_{\text{ssl}}$. We use the final layer of WavLM, and interpolate SSL features to match the temporal resolution

of our model. Formally, assuming $\cos(x, y)$ is the cosine similarity between x and y,

$$\mathcal{L}_{\text{ssl}} = 1 - \cos(s, \hat{s}).$$

iii) Cosine loss between the speaker embedding produced by TitaNet-L and the output from the global encoder, denoted $\mathcal{L}_{\text{emb}}$. Assuming p is a learned linear projection,

$$\mathcal{L}_{\text{emb}} = 1 - \cos(g, p(\hat{g}))$$

Vocoder training uses the adversarial training recipe from Vocos (Siuzdak, 2024). We use a multi-period discriminator (Kong et al., 2020) and a multi-band multi-scale STFT discriminator (Jang et al., 2021). The vocoder loss is composed of multi-scale mel spectrogram loss (Lee et al., 2023), adversarial loss, and feature matching loss. The vocoder is trained on ground-truth mels.

Table 1: Notation used in our model.

Symbol	Description
m	Target log-mel spectrogram of the utterance
$\tilde{m} = D(m)$	Log-mel spectrogram after stochastic degradation D
s	Target SSL features from WavLM-large
g	Target speaker embedding from TitaNet-L
$\hat{g}$	Predicted embedding from the global encoder
$\hat{m}$	Predicted log-mel spectrogram from the acoustic decoder
$\hat{s}$	Predicted SSL features from the semantic decoder

We find that training the autoencoder and vocoder together from scratch leads to poor performance because the high reconstruction error early in training causes adversarial training collapse, consistent with Xue et al. (2025). Instead, we adopt a two-stage framework. In the first stage, we train the autoencoder and vocoder independently. Critically, the bottleneck-constrained autoencoder is initially trained without adversarial training, avoiding training divergence. After both models have converged, we then fine-tune both models in an end-to-end fashion. We find this vastly improves training stability and reconstruction quality.

4 Experiments

Training Setup. We train CleanCodec on the full training set of LibriTTS-R (Koizumi et al., 2023), consisting of 585 hours of read speech from 2,456 speakers, as well as a 1,800-hour subset from the Emilia-YODAS dataset (He et al., 2024), bringing the total dataset size to approximately 2,400 hours. To evaluate the scaling performance of CleanCodec, we train model variants with a downsampling factor of 1 and 2 (as opposed to 5), achieving token rates of 62.5 t/s and 31.25 t/s, respectively. We refer to these models as CleanCodec@62.5, CleanCodec@31.25, and CleanCodec@12.5. Additional training details can be found in Appendix A.

Baselines. We compare our method with a variety of recent SOTA audio codecs, including Mimi (Défossez et al., 2024), BiCodec (Wang et al., 2025), XCodec2 (Ye et al., 2025), WavTokenizer (Ji et al., 2025), FocalCodec (Libera et al., 2025), Qwen3-TTS-Tokenizer (Qwen3) (Team, 2026), and Kanade (Huang et al., 2026).

Evaluation Metrics. We evaluate reconstructed audio using the following metrics:

- **Speaker identity:** Speaker embedding cosine similarity (SIM) using the ReDimNet-M speaker verification model (Yakovlev et al., 2024). We use ReDimNet-M instead of the more common WavLM-SV because it is a stronger SV model and provides greater discriminability between baselines.
- **Reconstruction faithfulness:** L1 distance between log-mel spectrograms (Mel L1).
- **Speech intelligibility:** word/character error rate (WER/CER) using `parakeet-tdt-0.6B-v2` speech recognition model (Sekoyan et al., 2025). These metrics are computed relative to the Parakeet transcription of the original audio so that the ground truth has a WER/CER of 0.
- **Quality:** UTMOS (Saeki et al., 2022) and SI-SDR from the Torchaudio-Squim evaluation suite (Kumar et al., 2023a).

We also report the token rate of each audio codec. Smaller token rates compress audio more aggressively, so codecs with higher token rates should have better metrics. As a result, we classify our baselines by token rate and compare our method against baselines of similar token rate. Specifically, we compare CleanCodec@12.5 with baselines below 50 t/s and compare CleanCodec@62.5 and CleanCodec@31.25 with baselines above 50 t/s. We take care to ensure that our models generally have a lower token rate than the baselines so that our improvements cannot be confounded with using a higher token rate.

Table 2: Evaluation results on LibriTTS test-clean and test-other.

		LibriTTS test-clean						LibriTTS test-other					
	Token Rate (t/s)	WER (↓)	CER (↓)	SIM (↑)	UTMOS (↑)	SI-SDR (↑)	Mel L1 (↓)	WER (↓)	CER (↓)	SIM (↑)	UTMOS (↑)	SI-SDR (↑)	Mel L1 (↓)
GT	–	0.0	0.0	1.00	4.20	26.49	0.00	0.0	0.0	1.00	4.01	24.44	0.00
Token Rate > 100 t/s													
Qwen3	200	1.3	0.7	0.90	4.27	27.99	0.29	2.6	1.4	0.90	4.09	27.85	0.29
Token Rate ≥ 50 t/s													
Mimi	100	3.5	1.9	0.72	3.86	17.48	0.56	7.5	4.4	0.70	3.63	17.21	0.56
Qwen3	100	2.6	1.4	0.72	3.92	19.45	0.38	4.7	2.5	0.72	3.73	20.20	0.38
BiCodec	50	2.6	1.3	0.78	4.25	**26.49**	0.55	6.2	3.4	0.77	4.07	**26.14**	0.62
XCodec2	50	2.6	1.5	0.77	4.20	23.51	0.39	5.4	3.1	0.75	4.01	23.47	0.40
Qwen3	50	7.9	4.5	0.31	2.65	9.01	0.63	15.8	9.4	0.30	2.51	8.91	0.61
CleanCodec@62.5	62.5	**1.3**	**0.6**	**0.90**	4.23	22.94	**0.29**	**2.7**	**1.6**	**0.89**	4.06	22.83	**0.29**
CleanCodec@31.25	31.25	1.7	0.9	0.88	**4.27**	22.02	0.33	3.6	1.9	0.87	**4.10**	22.12	0.33
Token Rate < 50 t/s													
WavTokenizer	40	9.0	5.0	0.62	3.81	19.98	0.46	19.1	11.5	0.61	3.63	20.94	0.47
FocalCodec	12.5	8.3	4.5	0.45	4.21	**25.19**	0.78	16.8	9.7	0.42	4.05	**25.25**	0.79
Kanade	12.5	4.0	2.1	0.65	4.22	23.60	0.57	8.2	4.5	0.64	4.10	23.77	0.59
CleanCodec@12.5	12.5	**2.7**	**1.4**	**0.86**	**4.32**	21.96	**0.44**	**5.7**	**3.2**	**0.84**	**4.17**	21.73	**0.44**

Table 3: Evaluation results on OOD speech. Full results in Table 10.

	Token Rate (t/s)	WER (↓)	CER (↓)	SIM (↑)	UTMOS (↑)	SI-SDR (↑)	Mel L1 (↓)
Expresso							
GT	–	0.0	0.0	1.00	3.43	24.15	0.00
WavTokenizer	40	19.7	11.4	0.49	2.95	16.56	0.53
FocalCodec	12.5	21.0	11.3	0.34	**3.75**	22.53	0.89
Kanade	12.5	9.3	4.9	0.55	3.38	22.81	0.61
CleanCodec@12.5	12.5	**3.9**	**1.9**	**0.82**	3.55	**22.99**	**0.45**
AISHELL-3							
GT	–	–	0.0	1.00	2.68	21.78	0.00
WavTokenizer	40	–	6.1	0.50	2.45	18.90	0.52
FocalCodec	12.5	–	15.0	0.24	**3.68**	**25.43**	0.79
Kanade	12.5	–	7.5	0.47	3.18	24.45	0.58
CleanCodec@12.5	12.5	–	**1.5**	**0.84**	3.04	22.41	**0.39**
CML-TTS							
GT	–	0.0	0.0	1.00	3.03	22.55	0.00
WavTokenizer	40	29.6	14.8	0.58	2.71	20.51	**0.47**
FocalCodec	12.5	54.4	29.9	0.35	**3.58**	**24.94**	0.75
Kanade	12.5	42.1	21.4	0.57	3.42	23.98	0.59
CleanCodec@12.5	12.5	**12.1**	**5.1**	**0.78**	3.47	22.11	0.65

4.1 Speech Reconstruction

We evaluate all baselines on the test-clean and test-other splits of LibriTTS (Zen et al., 2019), containing 8.6 and 6.7 hours, respectively. Our results can be found in Table 2, with the strongest results in bold and the second-strongest underlined. CleanCodec@12.5 achieves either the first or second-strongest results in the vast majority of benchmarks, achieving a new SOTA result. Notably, CleanCodec@12.5 improves substantially over existing baselines in SIM (0.86 vs 0.65) and WER (2.7 vs 4.0) scores, even achieving competitive results with other codecs with token rates >50 t/s. This indicates that CleanCodec@12.5 achieves much better speaker reconstruction and speech intelligibility compared to other audio codecs.

Extending our methodology to higher token rates, we see that CleanCodec's performance consistently improves as token rate increases. CleanCodec@31.25 surpasses both BiCodec and XCodec2 in 5 of 6 benchmarks, despite these codecs using a 60% higher token rate. In addition, CleanCodec@62.5 achieves competitive performance with Qwen3-TTS-Tokenizer while having a 3x lower token rate.

Table 5: Effects of noise removal preprocessing on quality metrics.

Model	Denoised?	WER	CER	SIM	UTMOS	SI-SDR	Mel L1
FocalCodec	Yes	9.7	5.5	0.44	4.29	25.88	0.80
FocalCodec	No	8.3	4.5	0.45	4.21	25.19	0.78
Kanade	Yes	4.9	2.4	0.63	4.28	23.80	0.58
Kanade	No	4.0	2.1	0.65	4.22	23.60	0.57

To demonstrate the robustness of our model, we also evaluate on a variety of out-of-distribution datasets, including Expresso (Nguyen et al., 2023), AISHELL-3 (Shi et al., 2021), and CML-TTS (de Oliveira et al., 2023), covering expressive, multilingual, and in-the-wild speech. Table 3 contains OOD results, with full results in Appendix B due to page limit constraints. Despite not explicitly being trained for any of these domains, we find that CleanCodec achieves strong results compared to other audio codecs, maintaining or improving its lead against baselines. In particular, we find that audio codecs with low token rates exhibit significantly degraded speech intelligibility on OOD datasets. However, CleanCodec@12.5 experiences much less degradation, achieving competitive WER/CER in Expresso and AISHELL-3.

4.2 Denoising Effectiveness

To analyze CleanCodec's denoising performance, we use VoiceBank-DEMAND (Valentini-Botinhao et al., 2016), a speech enhancement dataset consisting of pairs of clean and noisy speech. We extract the noisy audio clips and measure the improvement in perceptual quality metrics after reconstruction. Our results are in Table 4, where we report standard metrics STOI, PESQ, and SI-SDR from Torchaudio-Squim.

Table 4: CleanCodec achieves the highest improvement in perceptual quality metrics on VoiceBank-DEMAND noisy audio.

	Token Rate	STOI	PESQ	SI-SDR
Original	–	0.960	2.31	8.58
FocalCodec	12.5	0.984	2.98	17.42
Kanade	12.5	0.971	2.55	11.29
CleanCodec	12.5	**0.996**	**3.84**	**26.12**

We note that even our baselines are able to uplift the speech quality somewhat, as their extremely low token rate forces the codecs to discard high-entropy noise. However, CleanCodec achieves the strongest results by a significant margin due to the proposed denoising training objective, demonstrating its superiority in encoding noisy audio.

We also test alternative methods to remove noise, such as preprocessing the input speech with a speech enhancement pipeline (Nakata et al., 2026). We apply this technique to our low-token-rate baselines to see whether their results can be improved.

Table 5 shows our results on LibriTTS test-clean using FocalCodec and Kanade. Although the improved UTMOS and SI-SDR scores indicate higher perceptual quality, all other metrics are degraded. We argue that it is insufficient to simply apply speech enhancement as a preprocessing step during inference. Even if the input speech is clean, the model has already been trained to allocate a certain portion of its information bottleneck towards encoding perceptually irrelevant information, wasting bits. By integrating the speech enhancement step into the training pipeline, we ensure that the bottleneck encodes relevant information.

4.3 Disentanglement Effectiveness

Disentangled codecs aim to separate audio into speaker characteristics (located in the global representation) and linguistic content (located in the local representation). To measure the disentanglement performance, we train models on (1) speaker verification performance using only the global representation and (2) speech recognition performance using only the local representation. For the speaker verification task, we directly use the global representation as the

Table 6: SV and ASR results for disentangled codecs.

	Token Rate (t/s)	SV ACC (↑)	SV EER (↓)	ASR WER (↓)	ASR CER (↓)
BiCodec	50	99.15	1.52	6.2	3.4
Kanade	12.5	96.82	3.38	8.1	4.0
CleanCodec@62.5	62.5	**99.99**	<u>0.23</u>	**4.7**	**2.6**
CleanCodec@31.25	31.25	<u>99.98</u>	**0.22**	<u>4.8</u>	<u>2.7</u>
CleanCodec@12.5	12.5	99.92	0.58	5.6	3.0

speaker embedding. We conduct speaker verification experiments using VoxCeleb-1 (Nagrani et al., 2017). We evaluate using equal error rate (EER) and accuracy (ACC) based on the closest speaker centroid. For speech recognition, we train decoder-only transformers on local audio tokens from LibriTTS. We evaluate using the LibriTTS test-clean split with WER and CER. We compare CleanCodec with our disentangled baselines: BiCodec and Kanade. We do not test against older disentangled codecs like FACodec (Ju et al., 2024) and TiCodec (Ren et al., 2024) because they have been shown to have noncompetitive performance compared to our baselines.

Our results can be found in Table 6. Our models achieve SOTA results in both tasks, indicating that CleanCodec most effectively separates linguistic and timbre information compared to our baselines. We attribute this to our global and local encoders with TitaNet and WavLM conditioning, respectively.

4.4 Downstream Tasks

We evaluate our model on two downstream tasks: i) **Voice conversion (VC):** Given a source clip and reference clip, convert the source clip to the reference voice while preserving the source linguistic content. As shown in Huang et al. (2026), non-disentangled codecs significantly underperform at VC tasks, so we compare our method against BiCodec and Kanade. To obtain voice-converted audio, we extract global and local representations from the source and reference audio. Then, we pass the global embedding from the reference audio and the local tokens from the source audio through the decoder to obtain the converted audio. We evaluate using VCTK (Yamagishi et al., 2019), a 40-hour dataset of parallel utterances from 109 speakers, using WER/CER (linguistic similarity), SIM (speaker similarity), and UTMOS (quality). ii) **Text-to-speech (TTS):** Given a reference clip and a text, generate audio of the reference voice speaking the given text. We train decoder-only transformers on local audio tokens from LibriTTS. We evaluate using Seed-TTS-eval (Anastassiou et al., 2024), and also report the training time and inference speed.

Table 7: VC results for disentangled codecs on VCTK.

	T. Rate (t/s)	SIM (↑)	UTMOS (↑)	WER (↓)	CER (↓)
GT	–	1.00	4.10	0.0	0.0
BiCodec	50	0.47	3.87	1.3	0.7
Kanade	12.5	0.52	**4.21**	2.2	1.1
CleanCodec	62.5	0.70	3.93	**0.8**	**0.3**
	31.25	<u>0.76</u>	3.95	<u>0.9</u>	<u>0.4</u>
	12.5	**0.81**	<u>4.09</u>	1.4	0.7

VC results can be found in Table 7. CleanCodec achieves the highest speaker similarity by a wide margin (0.81 for CleanCodec@12.5 vs 0.52 for Kanade). Although BiCodec achieved strong speaker similarity in the speech reconstruction task, we observed that changing the global encoding rarely changed the speaker's identity, indicating that BiCodec stores significant speaker information in the local audio tokens. As a result, it performed poorly on this task. CleanCodec also appears to exhibit this behavior, albeit to a lesser degree, as both speaker similarity and UTMOS scores moderately degrade as token rate increases. Nevertheless, it still achieves SOTA results in 3 of 4 metrics. TTS results can be found in Table 8. Here, the efficiency advantages of using a lower token rate become clear. Both training and inference speed are inversely proportional to the token rate, allowing a TTS model using CleanCodec to have $10\times$ faster training and $17\times$ faster inference than one using Qwen3-TTS-Tokenizer. In addition, using a lower token rate shortens training sequences, enabling the model to better model long-range audio dependencies. These factors combine for CleanCodec to achieve SOTA results compared to baselines, both in terms of efficiency and performance. This indicates that the audio tokens produced by CleanCodec contain rich information for speech synthesis.

4.5 Efficiency Analysis

We also conduct a thorough efficiency analysis of CleanCodec, measuring its inference speed as well as how efficiently it utilizes its codebook. We report this analysis in Appendix C due to space constraints.

4.6 Ablation Study

We conduct an extensive ablation study to validate our design choices, with results in Table 9:

Denoising objective: We train an otherwise identical model without degraded-input/clean-target supervision. Removing the proposed objective degrades five of six metrics, showing that denoising improves how the bottleneck prioritizes perceptually important speech information.

Table 8: TTS results on Seed-TTS-eval.

	Token Rate (t/s)	Train Time (hh:mm, ↓)	RTF (RTFx) (↓, ↑)	SIM (↑)	WER (↓)
Qwen3	200	10:02	2.930 (0.3x)	0.36	9.1
Mimi	100	05:11	1.452 (0.7x)	0.34	10.8
XCodec2	50	02:44	0.688 (1.5x)	0.40	<u>4.9</u>
BiCodec	50	02:49	0.691 (1.4x)	<u>0.51</u>	5.2
WavTokenizer	40	02:19	0.552 (1.8x)	0.37	16.6
FocalCodec	12.5	00:58	0.172 (5.8x)	0.31	12.2
Kanade	12.5	00:59	0.169 (5.9x)	0.45	5.6
CleanCodec	12.5	01:00	0.170 (5.9x)	**0.56**	**3.9**

Table 9: Ablation study.

	WER (↓)	CER (↓)	SIM (↑)	UTMOS (↑)	SI-SDR (↑)	Mel L1 (↓)
Baseline	2.7	1.4	0.86	4.32	21.96	0.44
No denoise	3.4	1.7	0.82	4.24	20.55	0.41
One-stage training	4.1	2.2	0.80	4.01	18.58	0.57
HuBERT SSL cond.	2.9	1.4	0.85	4.32	22.03	0.45
No SSL cond.	8.1	4.3	0.84	4.34	22.56	0.42
WavLM global cond.	2.9	1.5	0.77	4.32	21.55	0.44
No global cond.	2.8	1.5	0.74	4.33	21.88	0.43
Kanade training setup	3.2	1.7	0.78	4.22	20.99	0.50
Kanade	*4.0*	*2.1*	*0.65*	*4.22*	*23.60*	*0.57*

Two-stage training: We jointly train the autoencoder and vocoder from scratch, instead of training each separately followed by end-to-end fine-tuning. This causes significant degradation in all metrics. Early in training, we find the model struggles to fool the discriminators, causing high reconstruction loss that never recovers after training.

SSL conditioning: We experiment with removing the SSL reconstruction branch from our decoder. Although there is a slight improvement in non-linguistic quality metrics like UTMOS and SI-SDR, there is a sharp drop in linguistic metrics, with a 3x increase in WER and CER. We also replace WavLM-large with HuBERT-large as the pretrained SSL model. We see that performance is very similar, indicating that CleanCodec is not sensitive to the choice of SSL model.

Global conditioning: We remove the global conditioning loss from the TitaNet-L speaker verification model. This causes a drop in speaker similarity without significant changes to other metrics. We also replace TitaNet with WavLM-SV, and find that this harms speaker reconstruction, corroborating our claim that WavLM-SV is subpar for sufficiently discriminating between speakers.

Training setup: We modify our training recipe to match Kanade, our strongest baseline. We shrink the vocoder to a 12-layer, 768-dim Vocos model to match Kanade's parameter count, and reduce the codebook and global embedding size to match Kanade's bitrate. Finally, we train only using LibriTTS. Our resulting model ties or outperforms the original Kanade model in 5 of 6 benchmarks, demonstrating that CleanCodec's benchmark improvements are mainly caused by improved training techniques, while model and dataset differences contribute additional gains.

5 Conclusion

We propose a denoising training objective for neural audio codecs that encourages the codec bottleneck to prioritize perceptually important speech information. We then introduce CleanCodec, a denoising disentangled audio codec that utilizes this objective together with auxiliary SSL and speaker-conditioning losses. The resulting codec achieves state-of-the-art performance in reconstruction, denoising, and downstream tasks at just 12.5 tokens per second.

AI USE STATEMENT

In this work, we used generative AI tools for editing the manuscript to improve readability. We have not used generative AI tools to generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, assist with translation, clean and reformat dataset, support qualitative and thematic data analysis, or interpret results. We have reviewed all AI-assisted work by manually checking and integrating all LLM-proposed edits to the manuscript. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

As stated in our abstract footnote, we will release models, code, and listening samples from our experiments on GitHub upon acceptance. We have also provided extensive details about our training setup in Appendix A, including model sizes, dataset configurations, optimizer hyperparameters, and hardware. Additional information about specific experiments can be found in their corresponding experiment subsections in Section 4.

REFERENCES

Philip Anastassiou, Jiawei Chen, Jitong Chen, Yuanzhe Chen, Zhuo Chen, Ziyi Chen, Jian Cong, Lelai Deng, Chuang Ding, Lu Gao, Mingqing Gong, Peisong Huang, Qingqing Huang, Zhiying Huang, Yuanyuan Huo, Dongya Jia, Chumin Li, Feiya Li, Hui Li, Jiaxin Li, Xiaoyang Li, Xingxing Li, Lin Liu, Shouda Liu, Sichao Liu, Xudong Liu, Yuchen Liu, Zhengxi Liu, Lu Lu, Junjie Pan, Xin Wang, Yuping Wang, Yuxuan Wang, Zhen Wei, Jian Wu, Chao Yao, Yifeng Yang, Yuanhao Yi, Junteng Zhang, Qidi Zhang, Shuo Zhang, Wenjie Zhang, Yang Zhang, Zilin Zhao, Dejian Zhong, and Xiaobin Zhuang. Seed-tts: A family of high-quality versatile speech generation models. *CoRR*, abs/2406.02430, 2024. doi: 10.48550/ARXIV.2406.02430. URL https://doi.org/10.48550/arXiv.2406.02430.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/92d1e1eb1cd6f9fba3227870bb6d7f07-Abstract.html.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, Xiangzhan Yu, and Furu Wei. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE J. Sel. Top. Signal Process.*, 16(6):1505–1518, 2022. doi: 10.1109/JSTSP.2022.3188113. URL https://doi.org/10.1109/JSTSP.2022.3188113.

Sanyuan Chen, Chengyi Wang, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, Lei He, Sheng Zhao, and Furu Wei. Neural codec language models are zero-shot text to speech synthesizers. *IEEE Transactions on Audio, Speech and Language Processing*, 33:705–718, 2025. doi: 10.1109/TASLPRO.2025.3530270.

Yongjian Chen, Tao Guan, and Cheng Wang. Approximate nearest neighbor search by residual vector quantization. *Sensors*, 10(12):11259–11273, 2010. doi: 10.3390/S101211259. URL https://doi.org/10.3390/s101211259.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/94b472a1842cd7c56dcb125fb2765fbd-Abstract-Conference.html.

Frederico Santos de Oliveira, Edresson Casanova, Arnaldo Cândido Júnior, Anderson da Silva Soares, and Arlindo R. Galvão Filho. CML-TTS: A multilingual dataset for speech synthesis in low-resource languages. In Kamil Ekstein, Frantisek Pártl, and Miloslav Konopík (eds.), *Text, Speech, and Dialogue - 26th International Conference, TSD 2023, Pilsen, Czech Republic, September 4-6, 2023, Proceedings*, volume 14102 of *Lecture Notes in Computer Science*, pp. 188–199. Springer, 2023. doi: 10.1007/978-3-031-40498-6_17. URL https://doi.org/10.1007/978-3-031-40498-6_17.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. *Trans. Mach. Learn. Res.*, 2023, 2023. URL https://openreview.net/forum?id=ivCd8z8zR2.

Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. Moshi: a speech-text foundation model for real-time dialogue. *CoRR*, abs/2410.00037, 2024. doi: 10.48550/ARXIV.2410.00037. URL https://doi.org/10.48550/arXiv.2410.00037.

Eduardo Fonseca, Xavier Favory, Jordi Pons, Frederic Font, and Xavier Serra. FSD50K: an open dataset of human-labeled sound events. *IEEE ACM Trans. Audio Speech Lang. Process.*, 30: 829–852, 2022. doi: 10.1109/TASLP.2021.3133208. URL https://doi.org/10.1109/TASLP.2021.3133208.

Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5-9, 2017*, pp. 776–780. IEEE, 2017. doi: 10.1109/ICASSP.2017.7952261. URL https://doi.org/10.1109/ICASSP.2017.7952261.

Robert M. Gray. Vector quantization. *IEEE Assp Magazine*, 1(2):4–29, 1984.

Haorui He, Zengqiang Shang, Chaoren Wang, Xuyuan Li, Yicheng Gu, Hua Hua, Liwei Liu, Chen Yang, Jiaqi Li, Peiyang Shi, Yuancheng Wang, Kai Chen, Pengyuan Zhang, and Zhizheng Wu. Emilia: An extensive, multilingual, and diverse speech dataset for large-scale speech generation. In *IEEE Spoken Language Technology Workshop, SLT 2024, Macao, December 2-5, 2024*, pp. 885–890. IEEE, 2024. doi: 10.1109/SLT61566.2024.10832365. URL https://doi.org/10.1109/SLT61566.2024.10832365.

Hyunjun Heo, Ui-Hyeop Shin, Ran Lee, Youngju Cheon, and Hyung-Min Park. Next-tdnn: Modernizing multi-scale temporal convolution backbone for speaker verification. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2024, Seoul, Republic of Korea, April 14-19, 2024*, pp. 11186–11190. IEEE, 2024. doi: 10.1109/ICASSP48485.2024.10447037. URL https://doi.org/10.1109/ICASSP48485.2024.10447037.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE ACM Trans. Audio Speech Lang. Process.*, 29:3451–3460, 2021. doi: 10.1109/TASLP.2021.3122291. URL https://doi.org/10.1109/TASLP.2021.3122291.

Zhijie Huang, Stephen McIntosh, Daisuke Saito, and Nobuaki Minematsu. Kanade: A simple disentangled tokenizer for spoken language modeling. *CoRR*, abs/2602.00594, 2026. doi: 10.48550/ARXIV.2602.00594. URL https://doi.org/10.48550/arXiv.2602.00594.

Won Jang, Dan Lim, Jaesam Yoon, Bongwan Kim, and Juntae Kim. Univnet: A neural vocoder with multi-resolution spectrogram discriminators for high-fidelity waveform generation. In Hynek Hermansky, Honza Cernocký, Lukás Burget, Lori Lamel, Odette Scharenborg, and Petr Motlícek (eds.), *22nd Annual Conference of the International Speech Communication Association, Interspeech 2021, Brno, Czechia, August 30 - September 3, 2021*, pp. 2207–2211. ISCA, 2021. doi: 10.21437/INTERSPEECH.2021-1016. URL https://doi.org/10.21437/Interspeech.2021-1016.

Shengpeng Ji, Ziyue Jiang, Wen Wang, Yifu Chen, Minghui Fang, Jialong Zuo, Qian Yang, Xize Cheng, Zehan Wang, Ruiqi Li, Ziang Zhang, Xiaoda Yang, Rongjie Huang, Yidi Jiang, Qian Chen, Siqi Zheng, and Zhou Zhao. Wavtokenizer: an efficient acoustic discrete codec tokenizer for audio language modeling. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=yBlVlS2Fd9.

Zeqian Ju, Yuancheng Wang, Kai Shen, Xu Tan, Detai Xin, Dongchao Yang, Eric Liu, Yichong Leng, Kaitao Song, Siliang Tang, Zhizheng Wu, Tao Qin, Xiangyang Li, Wei Ye, Shikun Zhang, Jiang Bian, Lei He, Jinyu Li, and Sheng Zhao. Naturalspeech 3: Zero-shot speech synthesis with factorized codec and diffusion models. In Ruslan Salakhutdinov, Zico Kolter, Katherine A. Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*, Proceedings of Machine Learning Research, pp. 22605–22623. PMLR / OpenReview.net, 2024. URL https://proceedings.mlr.press/v235/ju24b.html.

Yuma Koizumi, Heiga Zen, Shigeki Karita, Yifan Ding, Kohei Yatabe, Nobuyuki Morioka, Michiel Bacchiani, Yu Zhang, Wei Han, and Ankur Bapna. Libritts-r: A restored multi-speaker text-to-speech corpus. In Naomi Harte, Julie Carson-Berndsen, and Gareth Jones (eds.), *24th Annual Conference of the International Speech Communication Association, Interspeech 2023, Dublin, Ireland, August 20-24, 2023*, pp. 5496–5500. ISCA, 2023. doi: 10.21437/INTERSPEECH.2023-1584. URL https://doi.org/10.21437/Interspeech.2023-1584.

Nithin Rao Koluguri, Taejin Park, and Boris Ginsburg. Titanet: Neural model for speaker representation with 1d depth-wise separable convolutions and global context. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022, Virtual and Singapore, 23-27 May 2022*, pp. 8102–8106. IEEE, 2022. doi: 10.1109/ICASSP43922.2022.9746806. URL https://doi.org/10.1109/ICASSP43922.2022.9746806.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/c5d736809766d46260d816d8dbc9eb44-Abstract.html.

Anurag Kumar, Ke Tan, Zhaoheng Ni, Pranay Manocha, Xiaohui Zhang, Ethan Henderson, and Buye Xu. Torchaudio-squim: Reference-less speech quality and intelligibility measures in torchaudio. In *IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2023, Rhodes Island, Greece, June 4-10, 2023*, pp. 1–5. IEEE, 2023a. doi: 10.1109/ICASSP49357.2023.10096680. URL https://doi.org/10.1109/ICASSP49357.2023.10096680.

Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-fidelity audio compression with improved RVQGAN. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023b. URL http://papers.nips.cc/paper_files/paper/2023/hash/58d0e78cf042af5876e12661087bea12-Abstract-Conference.html.

Sang-gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon. Bigvgan: A universal neural vocoder with large-scale training. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=iTtGCMDEzS_.

Hanzhao Li, Liumeng Xue, Haohan Guo, Xinfa Zhu, Yuanjun Lv, Lei Xie, Yunlin Chen, Hao Yin, and Zhifei Li. Single-codec: Single-codebook speech codec towards high-performance speech generation. In Itshak Lapidot and Sharon Gannot (eds.), *25th Annual Conference of the International Speech Communication Association, Interspeech 2024, Kos, Greece, September 1-5, 2024*. ISCA, 2024. doi: 10.21437/INTERSPEECH.2024-1559. URL https://doi.org/10.21437/Interspeech.2024-1559.

Jiaqi Li, Xiaolong Lin, Zhekai Li, Shixi Huang, Yuancheng Wang, Chaoren Wang, Zhenpeng Zhan, and Zhizheng Wu. Dualcodec: A low-frame-rate, semantically-enhanced neural audio codec for speech generation. In Odette Scharenborg, Catharine Oertel, and Khiet Truong (eds.), *26th Annual Conference of the International Speech Communication Association, Interspeech 2025, Rotterdam, The Netherlands, 17-21 August 2025*. ISCA, 2025. doi: 10.21437/INTERSPEECH.2025-468. URL https://doi.org/10.21437/Interspeech.2025-468.

Luca Della Libera, Francesco Paissan, Cem Subakan, and Mirco Ravanelli. Focalcodec: Low-bitrate speech coding via focal modulation networks. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2025, NeurIPS 2025, San Diego, CA, USA, December 2-7, 2025 / Mexico City, Mexico, November 30 - December 5, 2025*, 2025. URL http://papers.nips.cc/paper_files/paper/2025/hash/220e6dbed88faa83bd9f9e8fa359e43d-Abstract-Conference.html.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 11966–11976. IEEE, 2022. doi: 10.1109/CVPR52688.2022.01167. URL https://doi.org/10.1109/CVPR52688.2022.01167.

Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: VQ-VAE made simple. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=8ishA3LxN8.

Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. Voxceleb: A large-scale speaker identification dataset. In Francisco Lacerda (ed.), *18th Annual Conference of the International Speech Communication Association, Interspeech 2017, Stockholm, Sweden, August 20-24, 2017*, pp. 2616–2620. ISCA, 2017. doi: 10.21437/INTERSPEECH.2017-950. URL https://doi.org/10.21437/Interspeech.2017-950.

Wataru Nakata, Yuki Saito, Yota Ueda, and Hiroshi Saruwatari. Sidon: Fast and robust open-source multilingual speech restoration for large-scale dataset cleansing. In *ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 17617–17621, 2026. doi: 10.1109/ICASSP55912.2026.11460553.

Tu Anh Nguyen, Wei-Ning Hsu, Antony D'Avirro, Bowen Shi, Itai Gat, Maryam Fazel-Zarandi, Tal Remez, Jade Copet, Gabriel Synnaeve, Michael Hassid, Felix Kreuk, Yossi Adi, and Emmanuel Dupoux. Expresso: A benchmark and analysis of discrete expressive speech resynthesis. In Naomi Harte, Julie Carson-Berndsen, and Gareth Jones (eds.), *24th Annual Conference of the International Speech Communication Association, Interspeech 2023, Dublin, Ireland, August 20-24, 2023*, pp. 4823–4827. ISCA, 2023. doi: 10.21437/INTERSPEECH.2023-1905. URL https://doi.org/10.21437/Interspeech.2023-1905.

Koji Okabe, Takafumi Koshinaka, and Koichi Shinoda. Attentive statistics pooling for deep speaker embedding. In B. Yegnanarayana (ed.), *19th Annual Conference of the International Speech Communication Association, Interspeech 2018, Hyderabad, India, September 2-6, 2018*, pp. 2252–2256. ISCA, 2018. doi: 10.21437/INTERSPEECH.2018-993. URL https://doi.org/10.21437/Interspeech.2018-993.

Yu Pan, Lei Ma, and Jianjun Zhao. Promptcodec: High-fidelity neural speech codec using disentangled representation learning based adaptive feature-aware prompt encoders. *CoRR*, abs/2404.02702, 2024. doi: 10.48550/ARXIV.2404.02702. URL https://doi.org/10.48550/arXiv.2404.02702.

Julian D. Parker, Anton Smirnov, Jordi Pons, CJ Carr, Zack Zukowski, Zach Evans, and Xubo Liu. Scaling transformers for low-bitrate high-quality speech coding. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=4YpMrGfldX.

Zhiliang Peng, Jianwei Yu, Wenhui Wang, Yaoyao Chang, Yutao Sun, Li Dong, Yi Zhu, Weijiang Xu, Hangbo Bao, Zehua Wang, Shaohan Huang, Yan Xia, and Furu Wei. Vibevoice technical report. *CoRR*, abs/2508.19205, 2025. doi: 10.48550/ARXIV.2508.19205. URL https://doi.org/10.48550/arXiv.2508.19205.

Yong Ren, Tao Wang, Jiangyan Yi, Le Xu, Jianhua Tao, Chu Yuan Zhang, and Junzuo Zhou. Fewer-token neural speech codec with time-invariant codes. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2024, Seoul, Republic of Korea, April 14-19, 2024*, pp. 12737–12741. IEEE, 2024. doi: 10.1109/ICASSP48485.2024.10448454. URL https://doi.org/10.1109/ICASSP48485.2024.10448454.

Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. UTMOS: utokyo-sarulab system for voicemos challenge 2022. In Hanseok Ko and John H. L. Hansen (eds.), *23rd Annual Conference of the International Speech Communication Association, Interspeech 2022, Incheon, Korea, September 18-22, 2022*, pp. 4521–4525. ISCA, 2022. doi: 10.21437/INTERSPEECH.2022-439. URL https://doi.org/10.21437/Interspeech.2022-439.

Robin Scheibler, Eric Bezzam, and Ivan Dokmanic. Pyroomacoustics: A python package for audio room simulation and array processing algorithms. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2018, Calgary, AB, Canada, April 15-20, 2018*, pp. 351–355. IEEE, 2018. doi: 10.1109/ICASSP.2018.8461310. URL https://doi.org/10.1109/ICASSP.2018.8461310.

Monica Sekoyan, Nithin Rao Koluguri, Nune Tadevosyan, Piotr Zelasko, Travis M. Bartley, Nikolay Karpov, Jagadeesh Balam, and Boris Ginsburg. Canary-1b-v2 & parakeet-tdt-0.6b-v3: Efficient and high-performance models for multilingual ASR and AST. *CoRR*, abs/2509.14128, 2025. doi: 10.48550/ARXIV.2509.14128. URL https://doi.org/10.48550/arXiv.2509.14128.

Yao Shi, Hui Bu, Xin Xu, Shaoji Zhang, and Ming Li. AISHELL-3: A Multi-Speaker Mandarin TTS Corpus. In *Interspeech 2021*, pp. 2756–2760, 2021. doi: 10.21437/Interspeech.2021-755.

Hubert Siuzdak. Vocos: Closing the gap between time-domain and fourier-based neural vocoders for high-quality audio synthesis. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=vY9nzQmQBw.

Qwen Team. Qwen3-tts technical report. *CoRR*, abs/2601.15621, 2026. doi: 10.48550/ARXIV.2601.15621. URL https://doi.org/10.48550/arXiv.2601.15621.

Cassia Valentini-Botinhao, Xin Wang, Shinji Takaki, and Junichi Yamagishi. Investigating RNN-based speech enhancement methods for noise-robust Text-to-Speech. In *9th ISCA Workshop on Speech Synthesis Workshop (SSW 9)*, pp. 146–152, 2016. doi: 10.21437/SSW.2016-24.

Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 6306–6315, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/7a98af17e63a0ac09ce2e96d03992fbc-Abstract.html.

Xinsheng Wang, Mingqi Jiang, Ziyang Ma, Ziyu Zhang, Songxiang Liu, Linqin Li, Zheng Liang, Qixi Zheng, Rui Wang, Xiaoqin Feng, Weizhen Bian, Zhen Ye, Sitong Cheng, Ruibin Yuan, Zhixian Zhao, Xinfa Zhu, Jiahao Pan, Liumeng Xue, Pengcheng Zhu, Yunlin Chen, Zhifei Li, Xie Chen, Lei Xie, Yike Guo, and Wei Xue. Spark-tts: An efficient llm-based text-to-speech model with single-stream decoupled speech tokens. *CoRR*, abs/2503.01710, 2025. doi: 10.48550/ARXIV.2503.01710. URL https://doi.org/10.48550/arXiv.2503.01710.

Gordon Wichern, Joe Antognini, Michael Flynn, Licheng Richard Zhu, Emmett McQuinn, Dwight Crow, Ethan Manilow, and Jonathan Le Roux. Wham!: Extending speech separation to noisy environments. In Gernot Kubin and Zdravko Kacic (eds.), *20th Annual Conference of the International Speech Communication Association, Interspeech 2019, Graz, Austria, September*

15-19, 2019, pp. 1368–1372. ISCA, 2019. doi: 10.21437/INTERSPEECH.2019-2821. URL https://doi.org/10.21437/Interspeech.2019-2821.

Rongkun Xue, Yazhe Niu, Shuai Hu, Zixin Yin, Yongqiang Yao, and Jing Yang. Hh-codec: High compression high-fidelity discrete neural codec for spoken language modeling. *CoRR*, abs/2507.18897, 2025. doi: 10.48550/ARXIV.2507.18897. URL https://doi.org/10.48550/arXiv.2507.18897.

Ivan Yakovlev, Rostislav Makarov, Andrei Balykin, Pavel Malov, Anton Okhotnikov, and Nikita Torgashov. Reshape dimensions network for speaker recognition. In Itshak Lapidot and Sharon Gannot (eds.), *25th Annual Conference of the International Speech Communication Association, Interspeech 2024, Kos, Greece, September 1-5, 2024*. ISCA, 2024. doi: 10.21437/INTERSPEECH.2024-2116. URL https://doi.org/10.21437/Interspeech.2024-2116.

Junichi Yamagishi, Christophe Veaux, and Kirsten MacDonald. CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit (version 0.92), 2019.

Zhen Ye, Xinfa Zhu, Chi-Min Chan, Xinsheng Wang, Xu Tan, Jiahe Lei, Yi Peng, Haohe Liu, Yizhu Jin, Zheqi Dai, Hongzhan Lin, Jianyi Chen, Xingjian Du, Liumeng Xue, Yunlin Chen, Zhifei Li, Lei Xie, Qiuqiang Kong, Yike Guo, and Wei Xue. Llasa: Scaling train-time and inference-time compute for llama-based speech synthesis. *CoRR*, abs/2502.04128, 2025. doi: 10.48550/ARXIV.2502.04128. URL https://doi.org/10.48550/arXiv.2502.04128.

Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J. Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. Libritts: A corpus derived from librispeech for text-to-speech. In Gernot Kubin and Zdravko Kacic (eds.), *20th Annual Conference of the International Speech Communication Association, Interspeech 2019, Graz, Austria, September 15-19, 2019*, pp. 1526–1530. ISCA, 2019. doi: 10.21437/INTERSPEECH.2019-2441. URL https://doi.org/10.21437/Interspeech.2019-2441.

Xin Zhang, Dong Zhang, Shimin Li, Yaqian Zhou, and Xipeng Qiu. Speechtokenizer: Unified speech tokenizer for speech language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=AF9Q8Vip84.

A Training Hyperparameters

The local encoder and acoustic decoder both use a 16-layer, 768-dim ConvNeXt backbone with a depthwise kernel size of 27 and an MLP intermediate size of 3072. The semantic decoder is identical to the acoustic decoder, but with a kernel size of 53. The global encoder is an 8-layer, 512-dim ConvNeXt backbone with a depthwise kernel size of 7 and an MLP intermediate size of 1536. The vocoder is a 16-layer, 1536-dim ConvNeXt backbone with a depthwise kernel size of 5 and an MLP intermediate size of 4608. In total, the autoencoder contains 243M parameters and the vocoder contains 228M parameters, for a total model size of 471M parameters. However, the semantic decoder can be discarded after training, reducing parameter count to 396M parameters.

During the first training stage, we independently train the autoencoder and vocoder for 1M steps each. We use audio samples of length 230,400 samples (9.6 seconds) with a batch size of 4. All utterances under this length are repeated, then trimmed to the desired length. We also apply a high-pass filter to all audio samples with a cutoff frequency of 80 Hz, as well as peak normalization with an amplitude of 0.75. We optimize using AdamW ($\beta_1 = 0.8, \beta_2 = 0.9, \text{wd} = 0.01$) and a cosine learning rate schedule initialized at 1e-4. During the second stage, we decrease the learning rate to 2e-6 and jointly train the autoencoder and vocoder in an end-to-end fashion for 200K steps. All other hyperparameters remain the same. Training was conducted on 1 A100 80GB GPU for one week.

In downstream ASR and TTS tasks, we use 16-layer, 4-head, 512-dim LLaMA-style transformers with a total of 96M parameters. For ASR models, we use training sequences of the format <audio tokens><BOS><text><EOS>. Only text tokens are used for computing cross-entropy loss. For TTS models, we use training sequences of the format <prompt tokens><text><BOS><audio

tokens><EOS>. Only the audio tokens are used for computing cross-entropy loss. For disentangled codecs, we prepend the global embedding to the beginning of the training sequence. Following Copet et al. (2023) for multi-codebook codecs, we flatten tokens and use a combined vocabulary, as this achieves the strongest performance. In both ASR and TTS tasks, models are trained for 10K steps with a batch size of 2,400 seconds. We optimize the model using AdamW ($\beta_1 = 0.9, \beta_2 = 0.95, \text{wd} = 0.1$) and a cosine learning rate schedule initialized at 1e-3.

B Out-of-Distribution Reconstruction Results

See Table 10 for full OOD results. WER is omitted for Chinese, which lacks word-level segmentation. In addition, we use the multilingual ASR model `parakeet-tdt-0.6B-v3` (Sekoyan et al., 2025) to evaluate CML-TTS and `stt_zh_conformer_transducer_large` for AISHELL-3.

C Efficiency Analysis

C.1 Codebook efficiency

We compare the codebook utilization of CleanCodec using normalized entropy:

$$\text{Normalized Entropy} = -\frac{1}{\log N} \sum_{x=1}^{N} p(x) \log(p(x)).$$

Results are reported in Table 11. Our model has the second-highest normalized entropy, behind only BiCodec, indicating very efficient codebook usage.

C.2 Bitrate efficiency

We report the bitrate (excluding global embedding), global embedding size (if applicable), and other details of the codecs used in our paper in Table 12.

We note that global embeddings have a complex relationship with the bitrate. In particular, the global embedding uses a fixed number of bits regardless of audio length, so its impact on bitrate would vary as audio length changes. To avoid confusion, we used token rate instead of bitrate throughout the paper.

C.3 Codec inference efficiency

We report the encoding and decoding speed of all audio codecs in this work in Table 13. Although CleanCodec contains more parameters than most baselines, it still achieves competitive inference speed, with the lowest encoding RTF and third-lowest total RTF, making it a strong choice for lightweight applications.

Table 10: Full evaluation results on OOD speech.

	Token Rate (t/s)	WER (↓)	CER (↓)	SIM (↑)	UTMOS (↑)	SI-SDR (↑)	Mel L1 (↓)
Expresso							
GT	–	0.0	0.0	1.00	3.43	24.15	0.00
Token Rate $>$ 100 t/s							
Qwen3	200	1.5	0.6	0.89	3.50	25.38	0.29
Token Rate $\geq$ 50 t/s							
Mimi	100	5.5	3.0	0.63	3.07	15.94	0.58
Qwen3	100	2.9	1.4	0.68	3.23	21.15	0.38
BiCodec	50	4.5	2.4	0.74	3.42	**23.94**	0.53
XCodec2	50	4.3	2.1	0.73	3.34	22.88	0.41
Qwen3	50	13.0	7.3	0.25	2.22	10.22	0.65
CleanCodec@62.5	62.5	**1.9**	**0.9**	**0.87**	3.43	23.07	**0.31**
CleanCodec@31.25	31.25	2.4	1.2	0.86	**3.47**	22.87	0.35
Token Rate $<$ 50 t/s							
WavTokenizer	40	19.7	11.4	0.49	2.95	16.56	0.53
FocalCodec	12.5	21.0	11.3	0.34	**3.75**	22.53	0.89
Kanade	12.5	9.3	4.9	0.55	3.38	22.81	0.61
CleanCodec@12.5	12.5	**3.9**	**1.9**	**0.82**	3.55	**22.99**	**0.45**
AISHELL-3							
GT	–	–	0.0	1.00	2.68	21.78	0.00
Token Rate $>$ 100 t/s							
Qwen3	200	–	0.5	0.89	2.96	24.05	0.30
Token Rate $\geq$ 50 t/s							
Mimi	100	–	1.9	0.57	2.61	18.38	0.55
Qwen3	100	–	1.3	0.68	2.77	22.09	0.37
BiCodec	50	–	1.0	0.77	2.92	22.72	0.55
XCodec2	50	–	1.1	0.76	2.83	22.79	0.40
Qwen3	50	–	17.8	0.24	2.02	9.71	0.63
CleanCodec@62.5	62.5	–	**0.5**	**0.88**	2.89	**23.06**	**0.29**
CleanCodec@31.25	31.25	–	0.7	0.87	**2.97**	22.88	0.32
Token Rate $<$ 50 t/s							
WavTokenizer	40	–	6.1	0.50	2.45	18.90	0.52
FocalCodec	12.5	–	15.0	0.24	**3.68**	**25.43**	0.79
Kanade	12.5	–	7.5	0.47	3.18	24.45	0.58
CleanCodec@12.5	12.5	–	**1.5**	**0.84**	3.04	22.41	**0.39**
CML-TTS							
GT	–	0.0	0.0	1.00	3.03	22.55	0.00
Token Rate $>$ 100 t/s							
Qwen3	200	3.2	1.4	0.92	3.15	24.54	0.30
Token Rate $\geq$ 50 t/s							
Mimi	100	11.0	5.1	0.69	2.67	16.46	0.55
Qwen3	100	6.3	2.5	0.72	2.74	20.57	**0.37**
BiCodec	50	8.9	3.8	0.77	3.17	22.79	0.48
XCodec2	50	6.2	2.6	0.81	3.18	**23.16**	0.38
Qwen3	50	23.4	11.5	0.30	1.85	9.32	0.56
CleanCodec@62.5	62.5	**5.4**	**2.4**	**0.83**	**3.41**	22.86	0.50
CleanCodec@31.25	31.25	6.4	2.7	0.82	3.33	21.72	0.55
Token Rate $<$ 50 t/s							
WavTokenizer	40	29.6	14.8	0.58	2.71	20.51	**0.47**
FocalCodec	12.5	54.4	29.9	0.35	**3.58**	**24.94**	0.75
Kanade	12.5	42.1	21.4	0.57	3.42	23.98	0.59
CleanCodec@12.5	12.5	**12.1**	**5.1**	**0.78**	3.47	22.11	0.65

Table 11: Normalized entropy of different audio codecs. Baseline results reported from Huang et al. (2026).

Model	Entropy (↑)
BiCodec	**0.995**
SpeechTokenizer	0.984
Kanade	0.976
XCodec2	0.965
StableCodec	0.962
FACodec	0.953
DualCodec	0.923
Mimi	0.914
WavTokenizer	0.885
PAST	0.824
TiCodec	0.623
CleanCodec@12.5	<u>0.985</u>

Table 12: Architectural comparison of CleanCodec vs. baseline codecs.

Model	Token Rate (t/s)	Bitrate (bps)	Global size	Causal?
Qwen3	200	2200	–	✓
Mimi	100	1100	–	✓
Qwen3	100	1100	–	✓
CleanCodec	62.5	938	256	✗
XCodec2	50	800	–	✗
BiCodec	50	650	32	✗
Qwen3	50	550	–	✓
WavTokenizer	40	520	–	✗
CleanCodec	31.25	469	256	✗
CleanCodec	12.5	187	256	✗
Kanade	12.5	171	128	✗
FocalCodec	12.5	163	–	✗

Table 13: Codec encoding/decoding efficiency. Total RTF measures the speed of both encoding and decoding.

Model	Params	Enc RTF (RTFx) (↓, ↑)	Dec RTF (RTFx) (↓, ↑)	Total RTF (RTFx) (↓, ↑)
Qwen3	171M	0.0035 (284x)	0.0059 (169x)	0.0094 (106x)
Mimi	96M	0.0032 (317x)	0.0024 (415x)	0.0055 (180x)
BiCodec	156M	0.0061 (164x)	0.0024 (414x)	0.0085 (117x)
XCodec2	823M	0.0146 (68x)	0.0027 (368x)	0.0174 (57x)
WavTokenizer	81M	<u>0.0026</u> (379x)	<u>0.0012</u> (802x)	<u>0.0039</u> (257x)
FocalCodec	145M	0.0027 (376x)	**0.0009** (1123x)	**0.0036** (282x)
Kanade	207M	0.0035 (284x)	0.0021 (473x)	0.0056 (177x)
CleanCodec@12.5	396M	**0.0011** (928x)	0.0034 (291x)	0.0045 (221x)