AUV: TEACHING AUDIO UNIVERSAL VECTOR QUANTIZATION WITH SINGLE NESTED CODEBOOK

Yushen Chen[1,2], *Kai Hu*[3], *Long Zhou*[3], *Shulin Feng*[3], *Xusheng Yang*[4], *Hangting Chen*[3], *Xie Chen*[1,2†]

[1] X-LANCE Lab, Shanghai Jiao Tong University, China [2] Shanghai Innovation Institute, China
[3] Tencent Hunyuan, China [4] Peking University, Shenzhen, China

ABSTRACT

We propose AUV, a unified neural audio codec with a single codebook, which enables a favourable reconstruction of speech and further extends to general audio, including vocal, music, and sound. AUV is capable of tackling any 16 kHz mixed-domain audio segment at bit rates around 700 bps. To accomplish this, we guide the matryoshka codebook with nested domain-specific partitions, assigned with corresponding teacher models to perform distillation, all in a single-stage training. A conformer-style encoder-decoder architecture with STFT features as audio representation is employed, yielding better audio quality. Comprehensive evaluations demonstrate that AUV exhibits comparable audio reconstruction ability to state-of-the-art domain-specific single-layer quantizer codecs, showcasing the potential of audio universal vector quantization with a single codebook. The pre-trained model and demo samples are available at `https://swivid.github.io/AUV/`.

***Index Terms*—** Neural audio codec, Unified audio representation, Vector quantization, Audio coding

1. INTRODUCTION

Within the mainstream token-based language modeling paradigm for various speech and audio tasks [1–6], a compact discrete representation that supports faithful reconstruction and distinguishable generation continues to grow in importance. The key challenge of deriving neural audio codecs [7, 8] capable of producing such discrete tokens lies in two aspects: reconstruction quality and generative capacity. The token rate and codebook size inherently constrain an audio codec's reconstruction quality. Relaxing the information bottleneck with a higher token rate or a larger codebook benefits reconstruction straightaway. However, larger token rates and codebook sizes add to the burden on downstream tasks, as larger models and training resources are required; otherwise, the generation performance would be sacrificed, since the audio sequence is longer and the modeling space becomes more complicated.

Neural audio codecs based on Residual Vector Quantization (RVQ) [9–11] continue to push the limits of reconstruction fidelity and compactness. In contrast, recent works focused on developing single-layer quantizers [12–19] are catching up, and further facilitating autoregressive modeling as the computation complexity and generation latency are lower. On the other hand, researchers have carried out a considerable amount of exploration to encapsulate rich semantic information into discrete tokens produced by domain-specific codecs [5, 20–26]. Still, relatively fewer investigations have been made from a unified audio view, including speech, music, and sound, especially with a desired single codebook paradigm [27]. Therefore, we frame our work here on the tokenization of general audio, providing practical and usable representations to boost unified generation tasks.

In this work, we mainly focus on the tokenization process itself, investigating and consolidating the single-layer quantization method to achieve faithful reconstruction of general audio with high perceptual quality:

- We introduce **AUV**, achieve **A**udio **U**niversal **V**ector quantization with a single nested codebook. AUV is with a domain-specific matryoshka codebook serving as a weak prior, and receives further semantic guidance through distillation from corresponding domain teacher models.
- We opt to use conformer as the neural audio codec encoder and decoder, with the time-frequency domain feature STFT as audio representation. We derive this effective model design through preliminary experiments, providing insights into the pros and cons of architecture backbone choices.
- Comprehensive evaluations show that AUV excels existing unified codecs with a single codebook in reconstruction. Results on the speech synthesis task also showcase the superiority of our method, initially revealing the potential in generation.

2. RELATED WORK

Neural Audio Codecs A standard neural codec is an encoder-decoder model employing vector quantization [28] to the latent representation in the bottleneck. Discrete tokens from a finite codebook are typically extracted using the trained codec encoder and quantizer for downstream autoregressive generation tasks. Notably, most audio codecs are trained with the VQ-GAN framework [29]. From the research scope of unified single-codebook codecs, UniCodec [27] proposes a three-stage training with a rigid-split codebook and a domain-aware mixture-of-experts equipped encoder, surpassing its predecessor WavTokenizer [13], which directly uses a whole codebook for speech, music, and sound. However, UniCodec faces issues with complex multi-stage training, artifacts in reconstructed results, and a native deficiency in dealing with mixed-domain audio segments. Our work handles these challenges relatively better.

Semantic Audio Representation Learning Self-supervised learning (SSL) proficiency in speech-related tasks [30–32] has helped a lot with incorporating semantic information into discrete tokens compressed by acoustic neural audio codecs, facilitating speech language modeling. SpeechTokenizer [20] and Mimi [5] perform distill-based semantic injection with SSL-based speech representation models. X-codec [22] adopts a dual-encoder structure, leveraging a pretrained speech SSL model as the semantic branch to work with another acoustic encoder. UniCodec [27] introduces a

†Corresponding Author.

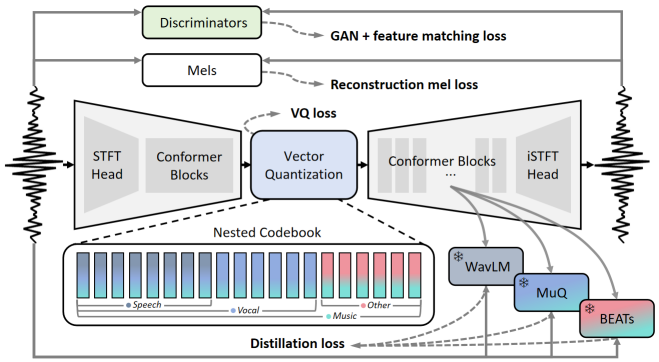

Fig. 1. Overview of AUV framework. During training, the input audio domain is made aware to the model, whereas it remains agnostic during inference, meeting the actual usage scenario.

self-supervised, masked modeling approach in a second training stage to enhance semantic information. In this work, we further take advantage of music and general audio foundation models [33, 34] pre-trained with self-supervised learning to provide distillation signals with distinct domain characteristics.

3. METHOD

In this section, we propose AUV, a single-layer acoustic audio codec incorporated with semantic distillation from a wide range of audio domains. We detail in the subsequent subsections.

- **Acoustic Neural Audio Codec**: The introduction of usual modules and the training procedure of an acoustic audio codec. Our model architecture and optimization strategies are also elaborated.
- **Nested Codebook for Universal Audio**: The design of a domain-specific matryoshka codebook for unified audio codecs.
- **Multi-domain Semantic Distillation**: The implementation of continuous representation distillation with teacher models to enrich the semantic information of our unified audio codec.

3.1. Acoustic Neural Audio Codec

As illustrated in Fig. 1, AUV builds upon the framework established by existing acoustic codecs, namely EnCodec [8], DAC [9], and so on. A single-codebook acoustic audio codec typically maintains an encoder-quantizer-decoder structure, the encoder transforms the input audio features into latent representations, the single-layer quantizer discretizes the latents into tokens, and the decoder finally reconstructs the waveform from the compressed discrete tokens.

Consistent with the findings of previous research [35], our early experiments reveal that scaling the encoder yields minimal gains for reconstruction while scaling the decoder boosts reconstruction. Furthermore, we empirically found that the convolutional structure with local perceptual capability is essential to single-stage VQ-GAN training; and directly replacing all convolution layers with transformer blocks suffers severe performance loss or may require dedicated design, e.g., a multi-stage training pipeline in MagiCodec [19].

Inspired by the success of Vocos [36] and Conformer [37] in various speech tasks, we propose to use conformer blocks as the encoder-decoder backbone and time-frequency domain feature STFT as the modeling target, with an STFT head to receive waveform input and an iSTFT head to recover the audio signal as output [36]. We weight more parameters to the codec decoder for the reason mentioned above, in practice, using more layers for the decoder compared to the encoder. The Fourier transform is configured with a chosen hop length to obtain spectrum features, with the same token rate desired for discrete tokens after vector quantization. Thus, there are no learnable upsampling or downsampling modules in the structure. By comparing with the highly competitive baseline, which is VQ with Mimi [5] encoder-decoder, we demonstrate the effectiveness of our model design. The details are provided in the experiments section.

The basic loss functions of AUV include quantizer loss, mel loss, adversarial loss, and feature matching loss, closely following BigCodec [14]. The distillation-related loss will be introduced in Sec. 3.3. A multi-period discriminator (MPD) from HiFi-GAN [38] and a multi-scale STFT (MS-STFT) discriminator are used for adversarial training, both in the way of BigCodec implementation. Notably, we incorporate an updated setting of FFT sizes {206, 334, 542, 876, 1418, 2296} from Stable-Codec [17] used for the MS-STFT discriminator. This set of settings is originally demonstrated by Stable-Codec to mitigate periodic artifacts in the spectrum, where our own experiments also show that it significantly improves the perceptual quality of reconstruction, especially in speaker similarity.

3.2. Nested Codebook for Universal Audio

The design of a nested codebook for an all-in-one unified audio tokenizer comes from the intuition that the domains of speech, vocal, music, and sound are not strictly separated or totally distinct. In fact, we consider vocal data, which is singing without accompaniment, to be a superset of speech. Similarly, music samples can be regarded as encompassing vocals. Under this assumption, we believe that unblocking and sharing the codebook to a certain extent can enhance its utilization rate. Meanwhile, we expect the nested way as a weak prior for the model, after training on diverse data, to learn to spontaneously select and place more semantically inclined information into the codebook partition assigned to speech, and the more rhythmic information into the partition solely for the vocal domain. The same applies to music. In addition, sound segments without human voices can be regarded as existing relatively independently.

In terms of implementation, our nested codebook will be attributed to four mutually overlapping domains. With a total codebook size of 16384, indices from 0 to 4095 are allocated for the speech domain, 0 to 8191 for vocal, 0 to 16383 for music, and 8192 to 16383 for other non-human sound. When scaling data, we further extend 4096 entries to the nested codebook for speech-related domains (0-8191, 0-12287, 0-20479, 12288-20479, correspondingly for speech, vocal, music, and other). Noted that the allocation of regions is empirical and should be flexible to different encoder-decoder architectures. During training, the input audio domain is provided to the model, while in inference, the model is domain-agnostic. The audio codec model is expected to rely solely on the learned encoder and quantizer to select the appropriate token from the entire codebook.

Ablation experiments have been made between three different codebook design choices: the no-split codebook (a whole), the rigid-split codebook (UniCodec-style partitioning [27]), and the nested codebook (ours). The experimental results in Sec. 4.4 show the superiority of our design, indicating a seamless integration of audio domains with AUV. And most importantly, index distribution probing statistics give a shred of strong evidence for our previous assumption that audio data from different domains are interrelated and mutually reinforcing.

3.3. Multi-domain Semantic Distillation

The idea of performing semantic distillation is straightforward, and the methods are well-established. However, unlike previous works that only focus on speech, we propose to leverage all domain teacher models to distill a unified audio codec model.

For continuous representation distillation, the training objective is to simultaneously minimize the $L1$ distance and maximize the cosine similarity between the D-dimensional hidden representations of the teacher's l-th layer $\boldsymbol{h}_t^{(l)}$ and the student's distillation learner head's prediction $\boldsymbol{s}_t$. Formally, the distillation loss is defined as:

$$\mathcal{L}_{distill}^{(l)} = \sum_{t=1}^{T} \left[\frac{1}{D} \left\| \boldsymbol{s}_t - \boldsymbol{h}_t^{(l)} \right\|_1 - \log \sigma \left(\cos \left(\boldsymbol{s}_t, \boldsymbol{h}_t^{(l)} \right) \right) \right],$$

where T is the number of frames, σ denotes sigmoid activation and $\cos(\cdot)$ indicates cosine similarity.

To be specific, we employ the average of the 13 to 24 layer representations of WavLM [32] as speech domain teacher; the average of the 5 to 10 layers of MuQ [34] as vocal and music domain teacher; and the average representation across all BEATs [33] layers as teacher for music and other (non-human sound) domains. The choice for MuQ is made based on a layer-wise investigation [39]. The student's distillation learner head is on the 6th layer output hidden state of our AUV decoder, ablated and explained in Sec. 4.4.

4. EXPERIMENTS

4.1. Experimental Setup

Datasets We train AUV on approximately 120,000 hours of data. For speech, we use 95K-hour Emilia [40] and LibriTTS [41]. For vocal and music, the in-house dataset is around 20K hours. For audio, we filter Audio Set [42] with labels to form a 4K-hour music set and an 800-hour non-human sound set. A 3K-hour mixed dataset is used for ablation, containing the 960h LibriSpeech *train* set, 600h vocal/music each randomly selected from in-house data, and the 800h no-human subset from Audio Set. The speech, vocal, and audio reconstruction performances are evaluated on LibriSpeech *test-clean* [43], an in-house test set, and Audio Set *eval*, respectively.

Pipeline All samples are resampled to 16 kHz with STFT hop length 320, resulting in a 50 Hz token rate. Segments longer than 12 seconds will be truncated during training. We use a global batch size of 128. The AdamW optimizer [44] has a peak learning rate of 1e-4, where the scheduler linearly warmed up for 5K updates, cosine decayed for 500K updates, and remained constant over the rest of the training. The hidden size of conformer blocks is 512, with an FFN multiply of 4. The codec encoder has 8 layers, and the decoder has 12 layers. We use 16384 and 20480 for codebook sizes (Sec. 3.2), and 8 for the factorized code dimension. The Exponential Moving Average (EMA) [45] weights are used for inference.

Baselines In the scope of unified audio codecs with a single codebook, UniCodec [27] is the major baseline. Other leading single-layer quantizer models are also compared, including BigCodec [14], X-codec2 [18], and MagiCodec [19].

Metrics In reconstruction quality evaluation, for speech, we report Word Error Rate (WER) from a HuBERT-based [31] ASR, Short Time Objective Intelligibility (STOI), Perceptual Evaluation of Speech Quality (PESQ), speaker similarity (SPK-SIM) from a WavLM-based [32] speaker verification model, and UTMOS [46]; for vocal, music, and other sound, we report the unified automatic quality assessment scores from Audiobox Aesthetics [47].

Table 1. Speech domain reconstruction evaluation results on LibriSpeech *test-clean*. TPS stands for token per second.

Model	Codebook Size	TPS	WER↓	STOI↑	PESQ-WB↑	SPK-SIM↑	UT MOS↑
Ground Truth	-	-	2.50	1.00	4.64	1.00	4.09
DAC	1024	50×12	2.61	0.97	4.01	0.95	4.00
BigCodec	8192	80	3.63	0.94	2.68	0.84	4.11
X-codec2	65536	50	3.20	0.92	2.43	0.82	4.12
MagiCodec	131072	50	4.25	0.92	2.54	0.77	4.17
UniCodec	16384	75	3.78	0.93	2.65	0.81	4.05
UniCodec$_{w/\ id}$	4096	75	4.14	0.92	2.55	0.81	4.03
AUV	20480	50	3.64	0.91	2.40	0.81	4.09

Table 2. Other domain reconstruction results. Audiobox Aesthetics [47] scores include: Content Enjoyment (CE), Content Usefulness (CU), Production Complexity (PC), Production Quality (PQ).

Model	Vocal test set				Audio Set *eval*			
	CE↑	CU↑	PC↑	PQ↑	CE↑	CU↑	PC↑	PQ↑
Ground Truth	5.69	6.04	3.44	6.81	4.52	5.73	4.10	6.33
UniCodec	5.06	5.44	2.66	6.44	4.09	5.21	4.03	5.88
AUV	**5.90**	**6.16**	**3.33**	**6.85**	**4.27**	**5.40**	**4.08**	**6.02**

4.2. Reconstruction Evaluation

Experimental results in Tab. 1 show that, as a unified acoustic audio codec, AUV is competitive with state-of-the-art domain-specific single-layer quantizer codecs in reconstruction quality. Our AUV has a lower token rate and modest codebook size, which has further advantages in usability. In Tab. 2, AUV comprehensively surpasses the previous leading model UniCodec, which further emphasizes that our design has great potential to serve as a general audio discrete representation to boost the development of downstream tasks.

Compared to UniCodec, AUV has single-stage training, is able to tackle mixed-domain audio samples, and lacks artifacts. As UniCodec relies on a domain-aware mixture-of-experts equipped encoder with a rigid-split codebook, it cannot switch freely during inference, thus performs poorly when dealing with audio that contains information from different domains. Regarding artifacts, UniCodec has significant issues with over-smoothing and aliasing on waveforms, suffers great degradation in human auditory perception, and performs weakly in terms of audio sample' dynamics. Fig. 2 presents a set of spectrum comparisons as an example.

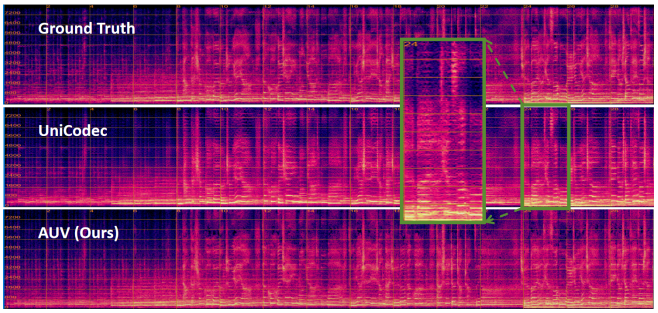

Fig. 2. Spectrograms of a snatch of music, including ground truth, and the reconstructed results by UniCodec and our AUV. UniCodec produces perceptible artifacts for audio samples other than speech.

Table 3. Ablation of model architecture, disillation method, and multi-domain codebook design on LibriSpeech *test-clean*.

ID	Enc-Dec Arch.	Trained Updates	Data Domain	Data Scale	Distilled Dec Layer	Codebook Type	Codebook Size	Reconstruction Quality					Code Index Distribution Ratio		
								WER↓	STOI↑	PESQ-WB↑	SPK-SIM↑	UT MOS↑	Speech	Vocal-other-than-Speech	Other
(A0)	Mimi	700K	speech	1K hrs	✗	-	4096	5.23	0.91	2.26	0.73	4.14	-	-	-
(A1)	AUV	400K	speech	1K hrs	✗	-	4096	4.60	0.90	2.18	0.76	4.15	-	-	-
(A2)	AUV	300K	speech	1K hrs	4	-	4096	4.64	0.90	2.13	0.73	4.15	-	-	-
(A3)	AUV	200K	speech	1K hrs	6	-	4096	4.39	0.90	2.14	0.76	4.05	-	-	-
(B0)	AUV	1M	all	3K hrs	✗	no split	16384	4.30	0.91	2.25	0.78	3.77	0.259	0.250	0.491
(B1)	AUV	1M	all	3K hrs	✗	rigid split	16384	4.21	0.91	2.31	0.79	3.86	0.322	0.214	0.464
	UniCodec	-	all	80K hrs	-	rigid split	16384	3.78	0.93	2.65	0.81	4.05	0.321	0.109	0.571
(B2)	AUV	1M	all	3K hrs	✗	nested	16384	3.99	0.91	2.35	0.80	3.89	0.371	0.189	0.441
(C0)	AUV	1M	all	3K hrs	6	nested	16384	4.05	0.91	2.28	0.80	3.96	0.382	0.193	0.424
(C1)	AUV	1M	all	3K hrs	6	nested	20480	4.00	0.91	2.32	0.80	3.95	0.515	0.142	0.343
(C2)	AUV	1M	all	120K hrs	6	nested	20480	3.64	0.91	2.40	0.81	4.09	0.591	0.109	0.300

Table 4. Speech domain generative capacity evaluation results on LibriSpeech-PC *test-clean*. TPS stands for token per second.

Used Codec	Multi-domain Distilled	Codebook Type	Codebook Size	TPS	WER↓	SPK-SIM↑	UT MOS↑
Ground Truth	-	-	-	-	1.86	0.69	4.09
BigCodec	-	-	8192	80	17.52	**0.47**	4.10
X-codec2	-	-	65536	50	12.87	0.43	4.19
UniCodec	-	rigid split	16384	75	12.79	0.38	4.17
(B0)	✗	no split	16384	50	5.45	0.43	4.15
(B1)	✗	rigid split	16384	50	6.26	0.43	4.20
(B2)	✗	nested	16384	50	4.99	0.44	4.27
(C0)	✓	nested	16384	50	**4.51**	0.44	4.26
(C2)	✓	nested	20480	50	4.89	0.43	**4.29**

4.3. Zero-shot TTS Results

To evaluate the generative capacity of AUV, we employ EmoVoice [48] as the text-to-speech (TTS) model to perform pure autoregressive stage training on the 1K-hour train set of LibriSpeech [43], and evaluate on a test subset of LibriSpeech derived from F5-TTS [49].

Results in Tab. 4 indicate the effectiveness of our nested codebook design and multi-domain distillation method, as (C0) and (C2) achieve much lower WER and higher UTMOS than models trained using other codec codes. (C0) has slightly better performance than (C2), and the reason for this can be attributed to the fact that the latter's larger codebook places higher demands on the modeling capabilities of downstream models (mentioned in Sec. 1), which may also be the cause of our failed training with MagiCodec extracted codes (its codebook size is 131072).

4.4. Ablation Study

Model Architecture Converged much faster, (A1) at a training update of 400K yields better WER, SPK-SIM, and UTMOS than the 700K-updated (A0) model. Thus, we opt to use conformer blocks with STFT as the backbone and conduct subsequent experiments.

Distillation Approach With a speech-only training, (A3) with a distillation learner on the 6th layer of the codec decoder stands out in WER compared to non-distilled and 4th-layer-distilled counterparts, (A1) and (A2) respectively. This result indicates that distillation is indeed beneficial for integrating semantic information into acoustic codecs, and it also shows that the reasonable selection of the student learner layer is crucial to maintain acoustic performance. Empirically, we found that a trivial probing test of attention rather than costly codec training can serve as a basis for selection; i.e., by discarding attention layers (of (A1)'s decoder) one by one and listening to the reconstructed speech, the layer that loses the most auditory quality due to discarding is thus found to be most correlated.

Codebook Design The reconstruction (in Tab. 3) and generation (in Tab. 4) capabilities of our proposed nested codebook are both verified ((B2) compared to no-split (B0) and rigid-split (B1)). The statistics of code index distribution highlight the rationality of considering the matryoshka codebook as a semantic prior, where (B2) attributes 37.1% of codes to the speech-only partition in the codebook when inferencing on speech-domain input samples, 1.5 times in terms of possibility than a random guess (25%). (C2)'s 59.1% is likewise 1.5 times that of random (40%, 8192 out of 20480). Moreover, the increase in the ratio from (B2) to (C0) adds to the effectiveness of multi-domain semantic distillation.

Data Scale Scaling the data has brought about a comprehensive gain in metrics ((C1) to (C2)). We are optimistic about further performance improvement if the data is further amplified.

5. CONCLUSION

In this work, we introduced AUV, an advanced all-in-one speech, vocal, music, and sound codec with a single-layer quantizer at 50 Hz. Rich semantic information is seamlessly integrated into our unified acoustic codec. This is achieved using conformers with STFT heads as backbone, a domain-adaptive nested codebook design as an effective semantic prior, and a multi-domain distillation leveraging self-supervised learned teacher models. Our evaluations demonstrate that AUV outperforms existing unified neural codecs with a single codebook in reconstruction and generation, and is competitive with state-of-the-art domain-specific codecs.

6. REFERENCES

[1] Zalán Borsos et al., "AudioLM: a language modeling approach to audio generation," *IEEE/ACM TASLP*, 2023.

[2] Eugene Kharitonov et al., "Speak, read and prompt: High-fidelity text-to-speech with minimal supervision," *TACL*, 2023.

[3] Sanyuan Chen et al., "Neural codec language models are zero-shot text to speech synthesizers," *IEEE TASLP*, 2025.

[4] Dongchao Yang et al., "UniAudio: An audio foundation model toward universal audio generation," in *Proc. ICML*, 2024.

[5] Alexandre Défossez, Laurent Mazaré, Manu Orsini, et al., "Moshi: a speech-text foundation model for real-time dialogue," *arXiv preprint arXiv:2410.00037*, 2024.

[6] Jin Xu, Zhifang Guo, Hangrui Hu, et al., "Qwen3-Omni Technical Report," *arXiv preprint arXiv:2509.17765*, 2025.

[7] Neil Zeghidour, Alejandro Luebs, et al., "SoundStream: An end-to-end neural audio codec," *IEEE/ACM TASLP*, 2021.

[8] Alexandre Défossez, Jade Copet, et al., "High fidelity neural audio compression," *arXiv preprint arXiv:2210.13438*, 2022.

[9] Rithesh Kumar, Prem Seetharaman, et al., "High-fidelity audio compression with improved RVQGAN," *in Proc. NIPS*, 2023.

[10] Simon Welker, Matthew Le, Ricky TQ Chen, et al., "FlowDec: A flow-based full-band general audio codec with high perceptual quality," in *Proc. ICLR*, 2025.

[11] Yang Yang, Yunpeng Li, George Sung, et al., "DiffSoundStream: Efficient speech tokenization via diffusion decoding," *arXiv preprint arXiv:2506.22362*, 2025.

[12] Hanzhao Li, Liumeng Xue, Haohan Guo, et al., "Single-codec: Single-codebook speech codec towards high-performance speech generation," in *Proc. Interspeech*, 2024.

[13] Shengpeng Ji, Ziyue Jiang, Wen Wang, et al., "WavTokenizer: an efficient acoustic discrete codec tokenizer for audio language modeling," in *Proc. ICLR*, 2025.

[14] Detai Xin et al., "BigCodec: Pushing the limits of low-bitrate neural speech codec," *arXiv preprint arXiv:2409.05377*, 2024.

[15] Yiwei Guo et al., "LSCodec: Low-bitrate and speaker-decoupled discrete speech codec," in *Proc. Interspeech*, 2025.

[16] Haibin Wu et al., "TS3-Codec: Transformer-based simple streaming single codec," in *Proc. Interspeech*, 2025.

[17] Julian D Parker et al., "Scaling transformers for low-bitrate high-quality speech coding," in *Proc. ICLR*, 2025.

[18] Zhen Ye, Xinfa Zhu, Chi-Min Chan, et al., "Llasa: Scaling train-time and inference-time compute for llama-based speech synthesis," *arXiv preprint arXiv:2502.04128*, 2025.

[19] Yakun Song, Jiawei Chen, et al., "MagiCodec: Simple masked gaussian-injected codec for high-fidelity reconstruction and generation," *arXiv preprint arXiv:2506.00385*, 2025.

[20] Xin Zhang et al., "SpeechTokenizer: Unified speech tokenizer for speech large language models," in *Proc. ICLR*, 2024.

[21] Haohe Liu et al., "SemantiCodec: An ultra low bitrate semantic audio codec for general sound," *IEEE JSTSP*, 2024.

[22] Zhen Ye, Peiwen Sun, Jiahe Lei, et al., "Codec does matter: Exploring the semantic shortcoming of codec for audio language model," in *Proc. AAAI*, 2025.

[23] Yaoxun Xu, Hangting Chen, et al., "MuCodec: Ultra low-bitrate music codec," *arXiv preprint arXiv:2409.13216*, 2024.

[24] Jiaqi Li, Xiaolong Lin, Zhekai Li, et al., "DualCodec: A low-frame-rate, semantically-enhanced neural audio codec for speech generation," in *Proc. Interspeech*, 2025.

[25] Yitian Gong, Luozhijie Jin, et al., "XY-Tokenizer: Mitigating the semantic-acoustic conflict in low-bitrate speech codecs," *arXiv preprint arXiv:2506.23325*, 2025.

[26] Yifan Yang et al., "Towards universal speech discrete tokens: A case study for ASR and TTS," in *Proc. ICASSP*, 2024.

[27] Yidi Jiang et al., "UniCodec: Unified audio codec with single domain-adaptive codebook," *in Proc. ACL*, 2025.

[28] Aaron Van Den Oord, Oriol Vinyals, et al., "Neural discrete representation learning," *in Proc. NIPS*, 2017.

[29] Patrick Esser, Robin Rombach, et al., "Taming transformers for high-resolution image synthesis," in *Proc. CVPR*, 2021.

[30] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *in Proc. NIPS*, 2020.

[31] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, et al., "HuBERT: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM TASLP*, 2021.

[32] Sanyuan Chen et al., "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE JSTSP*, 2022.

[33] Sanyuan Chen, Yu Wu, Chengyi Wang, et al., "BEATs: Audio pre-training with acoustic tokenizers," in *Proc. ICML*, 2023.

[34] Haina Zhu, Yizhi Zhou, Hangting Chen, et al., "MuQ: Self-supervised music representation learning with mel residual vector quantization," *arXiv preprint arXiv:2501.01108*, 2025.

[35] Yi-Chiao Wu et al., "AudioDec: An open-source streaming high-fidelity neural audio codec," in *Proc. ICASSP*, 2023.

[36] Hubert Siuzdak, "Vocos: Closing the gap between time-domain and fourier-based neural vocoders for high-quality audio synthesis," in *Proc. ICLR*, 2024.

[37] Anmol Gulati, James Qin, Chung-Cheng Chiu, et al., "Conformer: Convolution-augmented transformer for speech recognition," in *Proc. Interspeech*, 2020.

[38] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," in *Proc. NIPS*, 2020.

[39] Yizhi Zhou, Haina Zhu, and Hangting Chen, "Layer-wise investigation of large-scale self-supervised music representation models," *arXiv preprint arXiv:2505.16306*, 2025.

[40] Haorui He, Zengqiang Shang, Chaoren Wang, et al., "Emilia: An extensive, multilingual, and diverse speech dataset for large-scale speech generation," in *Proc. SLT*, 2024.

[41] Heiga Zen et al., "LibriTTS: A corpus derived from librispeech for text-to-speech," in *Proc. Interspeech*, 2019.

[42] Jort F Gemmeke et al., "Audio Set: An ontology and human-labeled dataset for audio events," in *Proc. ICASSP*, 2017.

[43] Vassil Panayotov et al., "LibriSpeech: an ASR corpus based on public domain audio books," in *Proc. ICASSP*, 2015.

[44] Ilya Loshchilov and Frank Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[45] Tero Karras et al., "Analyzing and improving the training dynamics of diffusion models," in *Proc. CVPR*, 2024.

[46] Takaaki Saeki et al., "UTMOS: Utokyo-sarulab system for voicemos challenge 2022," in *Proc. Interspeech*, 2022.

[47] Andros Tjandra, Yi-Chiao Wu, et al., "Meta Audiobox Aesthetics: Unified automatic quality assessment for speech, music, and sound," *arXiv preprint arXiv:2502.05139*, 2025.

[48] Guanrou Yang, Chen Yang, Qian Chen, et al., "EmoVoice: LLM-based emotional text-to-speech model with freestyle text prompting," in *Proc. ACM MM*, 2025.

[49] Yushen Chen et al., "F5-TTS: A fairytaler that fakes fluent and faithful speech with flow matching," in *Proc. ACL*, 2025.