A Survey of Neural Audio Codecs

Rushi Shah (B21AI032), Aditya Dhaduk (B21AI014)

February 2, 2025

Abstract

This survey presents a comprehensive analysis of neural audio codecs, examining their architectures, applications, and evaluation metrics. We explore how these modern approaches to audio compression leverage deep learning to achieve high-quality audio reconstruction at low bitrates. The survey covers major developments in the field, from basic compression-focused models to advanced systems that support real-time speech processing and natural conversations. We also discuss evaluation methodologies and identify current challenges and future research directions.

1 Motivation and Real-Life Applications

Audio compression has traditionally focused on reducing file sizes while maintaining perceptual quality [5, 6]. However, with the advent of deep learning and its success in various domains, neural audio codecs have emerged as a promising approach that goes beyond simple compression [1, 2]. These systems aim to address two fundamental challenges: efficient compression of high-dimensional audio data and the creation of discrete audio representations (or "tokens") that can interface with language models and other modalities.

In practical terms, neural audio codecs enable significant bandwidth savings, which is essential for real-time communication in bandwidth-constrained environments, such as mobile networks or remote conferencing platforms. They are also vital to natural-sounding speech synthesis, allowing systems to generate or reconstruct speech with greater clarity and expressiveness. Moreover, neural codecs support voice conversion and personalization, letting users adapt or transform voices in diverse applications - from entertainment to assistive technologies. Finally, by seamlessly integrating with language models and other modalities, neural audio codecs offer a unified modeling framework. This means that speech can be combined more easily with text and other data types, creating more sophisticated and interactive AI systems [3, 4].

2 History of Codecs

In the earliest stages, audio coding methods focused squarely on two main goals: compression (often referred to as bit-rate reduction) and reconstruction quality [9]. Researchers and engineers employed techniques like parametric modeling or filter banks to capture the essential

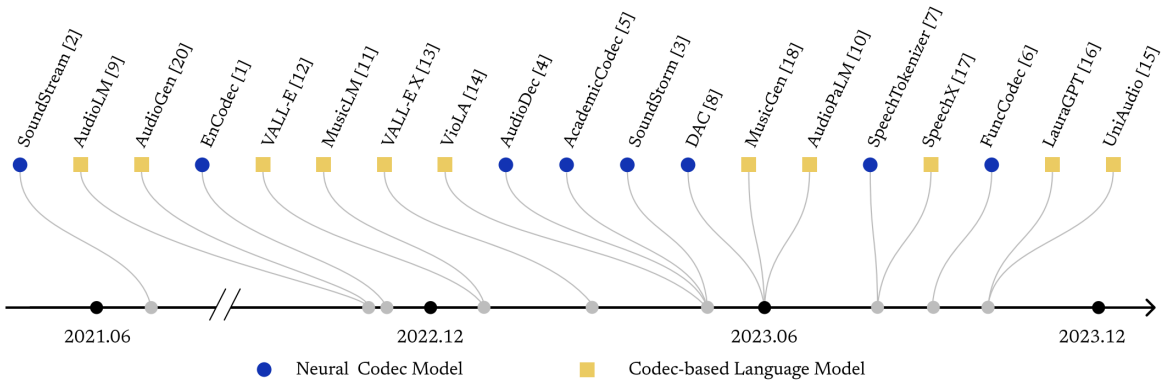

Figure 1: Credits: Towards audio language modeling - an overview

features of speech while minimizing the data needed for transmission or storage. Although these traditional codecs achieved practical successes - most notably in telecommunication systems [5, 6] - there was a growing need to handle richer and more dynamic speech content.

Over time, the increasing power of language modeling prompted a shift in how speech generation and processing were approached. Speech researchers began to discretize audio using neural networks, effectively turning speech into a sequence of learned "tokens." This transformation allowed speech-related tasks to be cast as language modeling problems, where the system could work through a pipeline of speech-to-text, text-to-text, and text-to-speech. While this pipeline proved valuable in many applications, it struggled to deliver fully natural, interactive conversations.

There were several reasons for these shortcomings. First, the pipeline-based approach compounded latency, as each component introduced delays. In a live interaction, this could result in global latencies of several seconds, undermining the fluidity of conversation. Second, relying solely on text meant that any non-written information - such as tone, emphasis, or emotional cues - remained invisible to the model. Finally, these systems were ill-equipped to handle real-world conversation features like interruptions, overlapping speech, or backchannelling.

Today, with the advent of large language models (LLMs) capable of remarkable text-based understanding and generation, the need for advanced neural audio codecs has become even more urgent [3]. Modern applications demand multi-turn, open-ended conversations that integrate speech and text seamlessly, without the delays and information loss of older pipelines. As a result, current efforts are focused on designing and refining codecs that can keep pace with these new, more interactive demands - maintaining efficiency, quality, and adaptability in the face of complex conversational scenarios.

3 How Neural Audio Codecs Work?

Neural audio codecs operate by transforming raw audio into a more compact, latent representation that captures the essential perceptual qualities of the signal [1, 2]. This transformation is typically carried out by a neural encoder network, which processes the input audio and

produces a set of latent features. These latent features are then quantized into discrete tokens, often through a technique called vector quantization [9]. By limiting the possible values of these tokens, the system effectively compresses the audio data into a smaller, more manageable form. Once the audio is represented in terms of these tokens, a decoder network reconstructs the audio signal, aiming to preserve both intelligibility and naturalness.

This process is usually trained end-to-end, meaning that during training, the encoder, quantizer, and decoder adjust their parameters collaboratively to minimize the difference between the original and reconstructed audio. The objective functions can include traditional signal-based losses (like mean squared error) and perceptual losses (which better reflect human judgments of audio quality). By carefully balancing these objectives, neural audio codecs can achieve high compression ratios without sacrificing the richness or clarity of the original sound.

3.1 Types of Audio Tokens

A key innovation in neural audio codecs is the use of different types of tokens to represent distinct layers of audio information. Specifically, researchers often separate the representation into semantic tokens and acoustic tokens, each serving a unique purpose in the generative process [3, 4].

3.1.1 Semantic Tokens

Semantic tokens capture the high-level content of speech - that is, what is being said. In this sense, they are similar to words or subwords in text-based language models. Because the focus is on representing the essential message, semantic tokens typically have a lower temporal resolution, summarizing larger chunks of audio into a compact sequence. They serve as the backbone for tasks like unconditioned speech generation, where determining the linguistic or phonetic structure is the first and most critical step. In a multi-stage generation pipeline (for example, in text-to-speech), semantic tokens are often produced first to outline the basic message before any finer details of the speech signal are filled in.

3.1.2 Acoustic Tokens

Acoustic tokens, on the other hand, capture the detailed, low-level aspects of the audio signal. This includes intonation, rhythm, timbre, and other features that contribute to a speaker's unique voice and delivery style [4]. Because they aim to reflect the intricate variations in audio over time, acoustic tokens are sampled more frequently than semantic tokens. This high temporal resolution makes them ideal for tasks where the quality and expressiveness of the output is paramount, such as conveying emotion or preserving a speaker's identity. When generating speech in a two-stage process, the system first decides on a sequence of semantic tokens, then leverages acoustic tokens to "paint in" the final sound, ensuring that the generated audio is both accurate in content and natural in delivery.

By combining these token types, modern neural audio codecs can produce speech that is not only intelligible and semantically correct but also rich in style, expression, and overall naturalness. This layered approach allows systems to handle the complexity of human

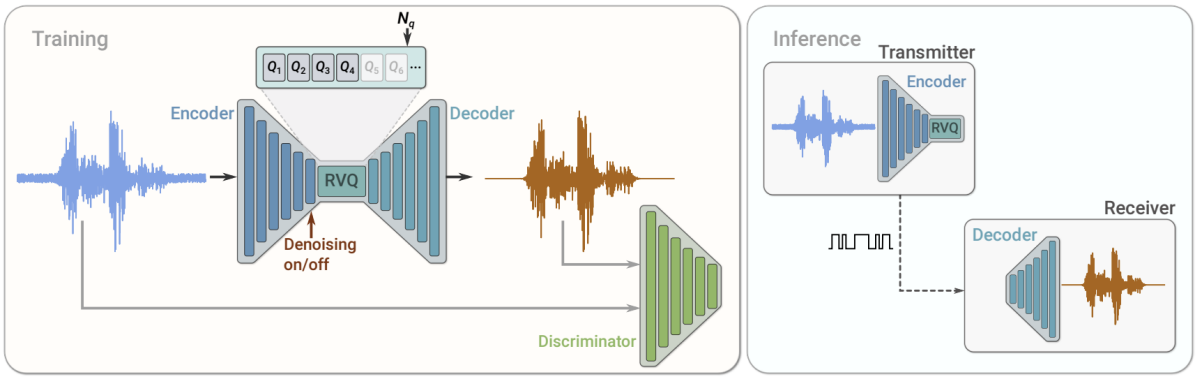

Figure 2: SoundStream: A convolutional encoder produces a latent representation of the input audio samples, which is quantized using a variable number n_q of residual vector quantizers (RVQ). During training, the model parameters are optimized using a combination of reconstruction and adversarial losses. An optional conditioning input can be used to indicate whether background noise has to be removed from the audio. When deploying the model, the encoder and quantizer on a transmitter client send the compressed bitstream to a receiver client that can then decode the audio signal.

speech more effectively, pushing the boundaries of what is possible in speech synthesis, voice conversion, and beyond [3].

4 Some Famous Codecs

4.1 SoundStream: An End-to-End Neural Audio Codec

SoundStream [1] is a neural audio codec that compresses raw audio into a compact representation while preserving clarity and detail. It uses a fully convolutional encoder-decoder design and is trained with both adversarial and reconstruction losses. This combination helps remove artifacts and improve the naturalness of the generated audio. At its core, SoundStream applies a multi-stage technique called Residual Vector Quantization (RVQ) to encode the latent representations. This approach allows the bitrate to be scaled from as low as 3 kbps up to 18 kbps. A training method known as quantizer dropout further enables the system to operate at different bitrates using the same model.

A key strength of SoundStream is its high compression efficiency. For example, at 3 kbps, it can surpass the Opus codec [5] running at 12 kbps and deliver results close to EVS [6] at 9.6 kbps. It also handles various audio types - speech, music, and general sounds - even at very low bitrates where traditional codecs typically fail. To support real-time applications, SoundStream employs causal convolutions, which minimize latency, making it suitable for tasks like live streaming on smartphones. It can also reduce background noise during compression, providing a combined denoising and encoding process without additional delays.

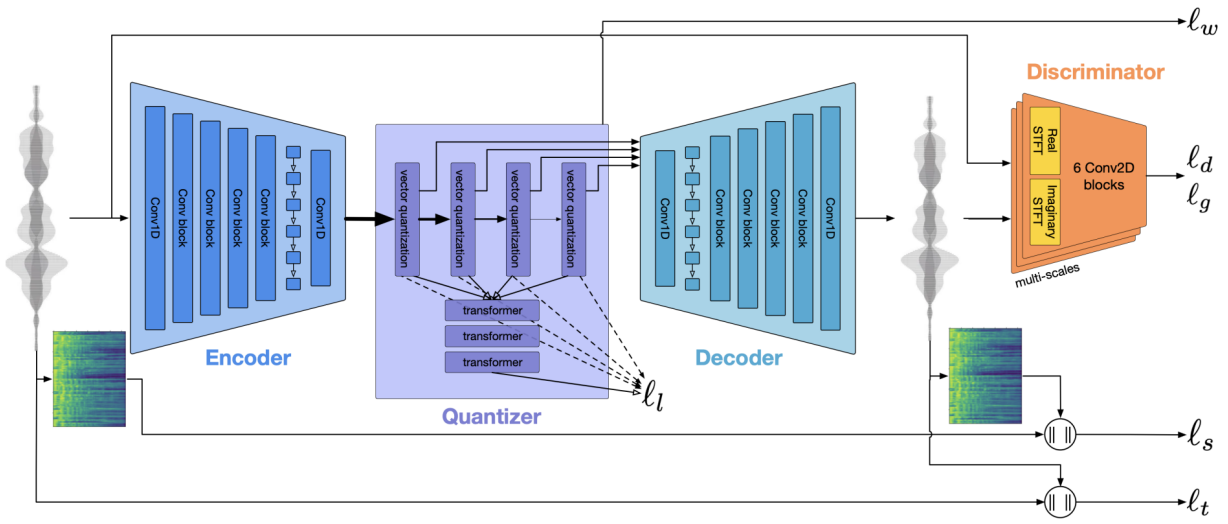

Figure 3: EnCodec: an encoder–decoder codec architecture which is trained with reconstruction (ℓ_f and ℓ_t) as well as adversarial losses (ℓ_g for the generator and ℓ_d for the discriminator). The residual vector quantization commitment loss (ℓ_w) applies only to the encoder. Optionally, a small Transformer language model for entropy coding is trained over the quantized units with ℓ_l, which reduces bandwidth even further.

4.2 EnCodec: High-Fidelity Neural Audio Compression

EnCodec [2] is a high-fidelity neural audio codec developed to achieve efficient, low-bitrate compression for both speech and music. Its design centers on three main components: an encoder, a residual vector quantization layer (RVQ), and a decoder. Trained end-to-end, EnCodec uses a combination of reconstruction loss, perceptual loss from adversarial discriminators, and a commitment loss that encourages stable and effective quantization [9].

The encoder is a streamable, convolution-based network that processes audio in either non-streamable or streamable modes. In the non-streamable mode, it handles short audio segments for higher fidelity, while the streamable mode employs causal padding for real-time processing. It downsamples the audio through strided convolutions and may include LSTM layers to capture sequential dependencies. The decoder then mirrors the encoder with transposed convolutions to upsample and reconstruct the final waveform.

EnCodec applies RVQ to iteratively refine the encoding, with each step targeting the residual error of the previous step. This method can be adjusted to different bitrates, often from 1.5 kbps to 24 kbps, allowing flexibility in balancing audio quality with bandwidth constraints. To push compression further, EnCodec uses a small Transformer-based language model to predict codebook entries and then applies entropy coding for an extra bandwidth reduction. In perceptual tests like MUSHRA [7], EnCodec shows better performance than well-known codecs such as Opus [5] and EVS [6], particularly at lower bitrates. It can also run in real time, with minimal latency, making it suitable for live streaming or interactive applications.

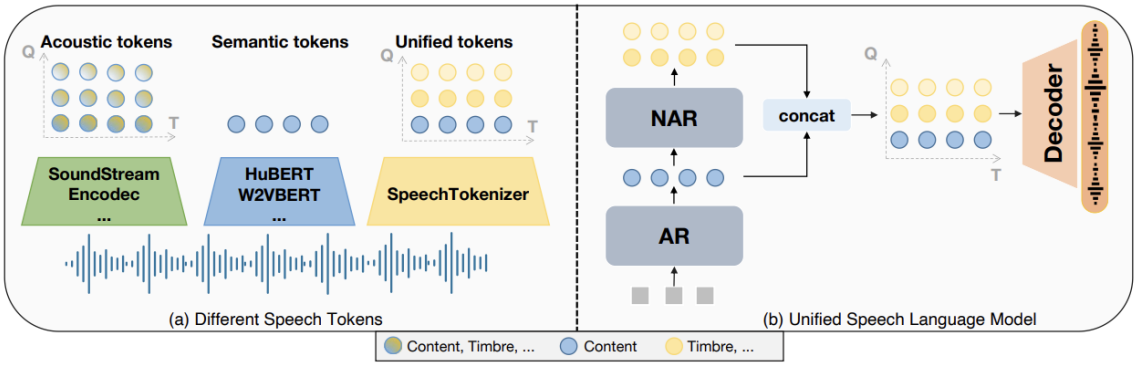

Figure 4: SpeechTokenizer: **Left** - Illustration of information composition of different discrete speech representations. **Right** - Illustration of unified speech language models. AR refers to autoregressive and NAR refers to non-autoregressive. Speech tokens are represented as colored circles and different colors represent different information.

4.3 SpeechTokenizer: Unified Speech Tokenizer for Speech Language Models

SpeechTokenizer [3] is a unified neural audio codec tailored for large-scale speech language models (SLMs). It integrates both semantic and acoustic information in a single framework, allowing for highly intelligible and high-fidelity speech generation without the need for separate token streams. Traditional approaches split speech into either semantic tokens, which capture "what is said" but lose acoustic detail, or acoustic tokens, which capture voice quality and style but can drift in content accuracy. SpeechTokenizer addresses both concerns by disentangling content and acoustic features hierarchically.

Its encoder-decoder architecture applies residual vector quantization in multiple layers. The first RVQ stage encodes semantic details that align closely with linguistic units (such as phonemes or words), while subsequent stages add fine acoustic characteristics like speaker style, pitch, and timbre. Training is done end-to-end, so the system learns to produce a set of tokens that preserve both the message and the expressive qualities of speech.

This codec is especially useful for speech language models that need to handle zero-shot text-to-speech tasks or generate speech without text supervision. In comparison with other methods like EnCodec [2], SpeechTokenizer often shows better word accuracy (lower WER) while remaining competitive in perceptual tests such as MUSHRA [7]. It also generalizes well to new languages, suggesting that its combined focus on semantic and acoustic layers can transfer across different linguistic settings.

4.4 Mimi

Mimi [4] is a neural audio codec developed as part of a real-time speech-to-speech model called Moshi. It produces discrete audio tokens that blend both semantic and acoustic information, making it well suited for low-latency speech generation. In contrast to older methods that require separate tokens for meaning and sound quality, Mimi's approach merges

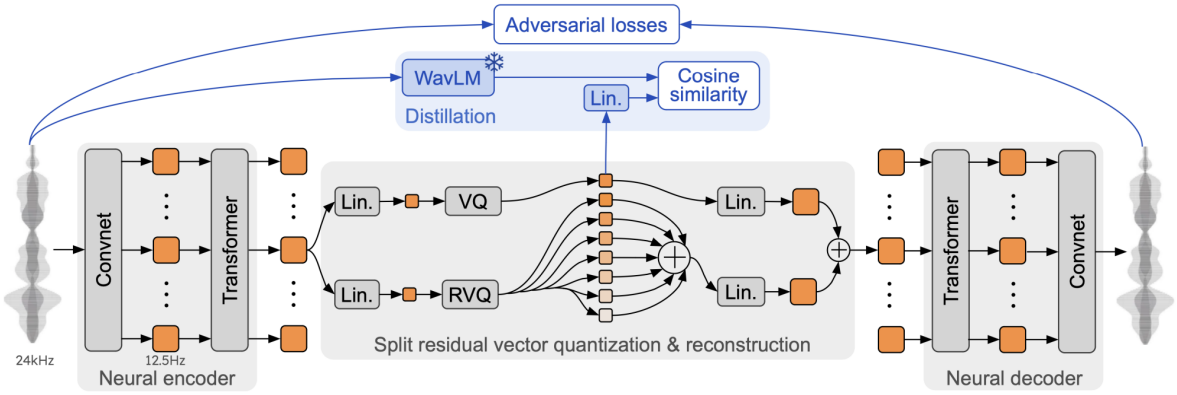

Figure 5: MIMI: During training (blue part, top), noncausal embeddings are distilled from WavLM into a single vector quantizer which produces semantic tokens, and is combined with separate acoustic tokens for reconstruction.

these aspects into a single token stream through a split residual vector quantization (RVQ) strategy.

The architecture of Mimi features a causal encoder based on SeaNet, which runs at 24 kHz and creates a latent representation at 12.5 frames per second. The first level of the quantizer encodes broad semantic content, while the remaining levels refine acoustic details such as pitch, rhythm, and timbre. By training the encoder to match embeddings from a self-supervised speech model, Mimi ensures that the first quantization layer captures meaningful linguistic information. The decoder then reconstructs the audio signal in a streaming-friendly way, making use of causal convolution or attention to minimize delay.

Evaluations show that Mimi outperforms other tokenizers in subjective quality tests while operating at a lower bitrate, around 1.1 kbps [4]. Although its phonetic scores may not reach the same level as some specialized speech encoders, Mimi balances intelligibility, audio fidelity, and semantic representation in a single codec. This design enables Moshi to perform text-free, interactive dialogue generation by relying exclusively on Mimi's integrated tokens.

Overall, these neural codecs - SoundStream [1], EnCodec [2], SpeechTokenizer [3], and Mimi [4] - demonstrate the fast-evolving landscape of audio compression. Each tackles the challenge of efficient encoding while maintaining intelligibility and naturalness, and each addresses the needs of different applications, from real-time streaming to large-scale speech language modeling.

5 Some Common Metrics

5.1 MUSHRA

MUSHRA (Multiple Stimuli with Hidden Reference and Anchor) is a standardized subjective listening test recommended by the ITU-R (BS.1534-3) for audio systems that introduce moderate quality impairments [7]. In a single test session, a listener is presented with

multiple versions of the same audio excerpt: one is the original reference, one is a hidden copy of the original, and one or more degraded versions generated by different codecs. The listener also hears "anchors," which are intentionally poor versions of the reference created by applying severe filtering or distortion.

Listeners rate each item on a scale typically ranging from 0 to 100, labeling them from bad to excellent. Since all samples can be compared directly, MUSHRA allows testers to discern subtle differences in audio quality across different codecs or conditions. However, MUSHRA can be susceptible to context bias if the presence of very poor anchors influences how listeners score mid-quality samples. It also requires careful participant screening, such as discarding data from listeners who fail to rate the hidden reference near the highest possible score in a substantial number of trials. Despite these challenges, MUSHRA remains a widely used standard for its balance between realism and practical test lengths.

5.2 VisQOL

VisQOL is an objective speech quality metric that predicts how listeners might perceive the clarity of a speech signal [8]. Instead of directly asking participants to score audio, VisQOL compares a reference signal with a degraded version and estimates a score that aligns with mean opinion scores (MOS) typically obtained from human listeners. The process begins by aligning and preprocessing the signals, creating spectrograms from both the reference and the degraded speech. VisQOL focuses on critical frequency ranges, such as narrowband (150 Hz to 3.4 kHz) or wideband (50 Hz to 8 kHz), to capture the most relevant information for speech intelligibility.

A central component of VisQOL is the Neurogram Similarity Index Measure (NSIM), which is an adaptation of a structural similarity measure originally used in image processing. NSIM compares the fine-grained, time-frequency details between the spectrograms of the reference and degraded signals. VisQOL then maps the resulting similarity score to a predicted MOS value using a nonlinear function that was calibrated against human listening tests. Because it requires a clean reference for comparison, VisQOL is classified as a full-reference metric. It is especially effective in VoIP contexts, where audio can suffer from packet loss, jitter, and other issues that make speech sound distorted or choppy. Although it may be less reliable in extremely poor audio conditions or in non-VoIP contexts, VisQOL provides a practical, automated way to assess speech quality and approximate how humans would judge the intelligibility and naturalness of a given speech sample.

6 Open Problems and Opportunities

Although neural audio codecs have made significant progress in both compression efficiency and audio quality, several challenges remain, offering promising directions for future research and development. First, models like Mimi [4] highlight a tradeoff between acoustic fidelity and semantic understanding. When a codec is too focused on capturing high-level semantic details, it can lose some of the subtle acoustic cues that make speech sound rich and realistic; conversely, emphasizing low-level acoustic fidelity might weaken the model's grasp of linguistic or phonetic content.

A second issue arises from the heavy reliance on Residual Vector Quantization (RVQ) [9]. While RVQ can represent audio tokens at different bitrates and scales, it suffers from discrete optimization problems that can lead to less-than-ideal code assignments. A related problem, codebook collapse, occurs when only a few entries in the quantization dictionary are used consistently, undermining the model's ability to learn a diverse range of audio features. Future work could investigate alternative quantization mechanisms or more robust training strategies to avoid these pitfalls.

Another open challenge is capturing high-resolution temporal dynamics in complex audio. Even though modern codecs have improved their ability to model rapid changes in speech or music, they can still fall short when it comes to very fast transient events, overlapping speech, or intricate sound textures. Approaches that combine fine-grained time-frequency analysis with high-capacity network architectures might help address these limitations.

Lastly, there is ample room for joint optimization and end-to-end training. Current systems often rely on multiple pre-trained modules - for example, encoders that focus on content and decoders that focus on style - without a unified training objective. Research into fully end-to-end methods could better balance semantic and acoustic goals, streamline inference pipelines, and reduce latency. By tackling these open problems, the field can continue to push the limits of what neural audio codecs can achieve in speech processing, music compression, and beyond.

References

[1] N. Zeghidour, O. Teboul, T. Sonnet, and M. Tagliasacchi, "SoundStream: An end-to-end neural audio codec," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2022 (extended from arXiv:2107.03312).

[2] A. Défossez, J. Copet, R. Caillon, G. Synnaeve, and Y. Adi, "High Fidelity Neural Audio Compression," in *Proc. 36th Conference on Neural Information Processing Systems (NeurIPS)*, 2022 (arXiv:2210.13438).

[3] W. Ao, K. Peng, X. Li, R. Pang, and T. Qin, "SpeechTokenizer: Unified speech tokenizer for speech large language models," *arXiv preprint arXiv:2305.11054*, 2023.

[4] Défossez, Alexandre and Mazaré, Laurent and Orsini, Manu and Royer, Amélie and Pérez, Patrick and Jégou, Hervé and Grave, Edouard and Zeghidour, Neil, "Moshi: a speech-text foundation model for real-time dialogue," *arXiv preprint arXiv:2410.00037*, 2024.

[5] J.-M. Valin, K. Vos, and T. Terriberry, "Definition of the Opus audio codec," *IETF RFC 6716*, 2012.

[6] 3GPP, "Technical specification group services and system aspects; 3GPP enhanced voice services (EVS) codec framework (3GPP TS 26.441 version 12.8.0 release 12)," 2015.

[7] ITU-R, "BS.1534-3: Method for the subjective assessment of intermediate quality levels of audio systems (MUSHRA)," International Telecommunication Union, Geneva, 2015.

[8] A. Hines, J. Skoglund, N. Harte, and A. Kokaram, "ViSQOL: The virtual speech quality objective listener," *The Journal of the Acoustical Society of America*, vol. 132, no. 5, pp. 3079–3084, 2012.

[9] R. M. Gray, "Vector quantization," *IEEE ASSP Magazine*, vol. 1, no. 2, pp. 4–29, 1984.