FuseCodec: Semantic-Contextual Fusion and Supervision for Neural Codecs

Anonymous ACL submission

Abstract

Speech tokenization enables discrete representation and facilitates speech language modeling. However, existing neural codecs capture low-level acoustic features, overlooking the semantic and contextual cues inherent to human speech. While recent efforts introduced semantic representations from self-supervised speech models or incorporated contextual representations from pre-trained language models, challenges remain in aligning and unifying the semantic and contextual representations. We introduce FuseCodec, which unifies acoustic, semantic, and contextual representations through strong cross-modal alignment and globally informed supervision. We propose three complementary techniques: (i) Latent Representation Fusion, integrating semantic and contextual features directly into the encoder latent space for robust and unified representation learning; (ii) Global Semantic-Contextual Supervision, supervising discrete tokens with globally pooled and broadcasted representations to enhance temporal consistency and cross-modal alignment; and (iii) Temporally Aligned Contextual Supervision, strengthening alignment by dynamically matching contextual and speech tokens within a local window for fine-grained token-level supervision. We further introduce FuseCodec-TTS, demonstrating our methodology's applicability to zero-shot speech synthesis. Empirically, FuseCodec achieves state-of-the-art performance in LibriSpeech, surpassing EnCodec, SpeechTokenizer, and DAC in transcription accuracy, perceptual quality, intelligibility, and speaker similarity. Results highlight the effectiveness of contextually and semantically guided tokenization for speech tokenization and downstream tasks.

1 Introduction

Tokenization has become foundational in natural language processing (NLP), enabling language models to learn discrete representations, while facilitating efficient autoregressive modeling and scalable downstream applications (Schmidt et al., 2024). Inspired by this paradigm, the speech domain has increasingly adopted neural codecs, popularized by Encodec (Défossez et al., 2022) and SoundStream (Zeghidour et al., 2022). Neural codecs tokenize speech using an encoder, residual vector quantizer, and decoder architecture, enabling modeling discrete representations suitable for modular extension to downstream tasks such as speech synthesis (Wang et al., 2023).

However, the continuous and multidimensional nature of human speech makes learning discrete representations inherently challenging (Ju et al., 2024). While neural codecs learn *acoustic representations* (waveform and low-level signal characteristics), they struggle to capture high-level semantics requiring downstream models to adopt additional self-supervised masked language objectives to derive *semantic representations* (phonetic content and linguistic meaning) (Borsos et al., 2023). To address this drawback, recent neural codec architectures incorporated semantic distillation from pretrained self-supervised speech models (Zhang et al., 2024; Défossez et al., 2024), improving the quality of speech reconstruction and the semantic aspect of learned representations.

In addition, another fundamental aspect of human speech remains missing in above mentioned works: speech is inherently grounded in context and surrounding cues (Brown et al., 2022). Discrete speech representations, lacking grounding in context, fall short of capturing this essential attribute (Hallap et al., 2023). While language models have demonstrated strong capabilities in learning such contextual dependencies from text corpora (Devlin et al., 2019a; Peters et al., 2018), speech tokenizers have yet to fully leverage these capabilities. Although a recent neural codec (Ahasan et al., 2024) explored matching discrete speech representations with contextual representation from a pre-trained

language model, it falls short in effective cross-modal alignment, constraining the model's ability to fully unify semantic and contextual information.

Therefore, despite recent advances, several challenges remain unaddressed. Firstly, current approaches fail to unify all three aspects of discrete speech representation: acoustic (learned by neural codecs), semantic (from self-supervised speech models), and contextual (from language models). Most work incorporates only semantic information (Zhang et al., 2024; Défossez et al., 2024; Ye et al., 2024), neglecting contextual grounding. Secondly, while a recent effort (Ahasan et al., 2024) attempts to integrate contextual representations, it lacks effective mechanisms for aligning text and speech modalities. Thirdly, existing methods rely on similarity-based objectives for representation matching without directly incorporating information into the latent space, limiting coherence and downstream performance. We address these challenges through our proposed methodologies, while preserving the core architecture and utilizing frozen representations with zero inference overhead.

To address these challenges, we propose a speech tokenization framework with three different strategies/variations that enrich discrete speech representations with unified and aligned semantic and contextual information. Our first strategy involves **(i) Latent Representation Fusion**, which integrates semantic and contextual embeddings into the encoder's latent space through cross-modal attention and additive fusion, resulting in more robust and coherent representations. Building on this, we present **(ii) Global Semantic-Contextual Supervision**, where globally pooled and broadcasted modality vectors supervise each quantized token across time, facilitating temporally consistent and globally informed representation learning. To enforce explicit alignment, we introduce another strategy: **(iii) Temporally Aligned Contextual Supervision**, which dynamically matches contextual and speech tokens prior to timestep-level similarity supervision, enabling fine-grained cross-modal alignment and enhancing representation quality.

Then, we instantiate our framework through three model variants: FuseCodec-Fusion with Latent Representation Fusion, FuseCodec-Distill with Global Semantic-Contextual Supervision, and FuseCodec-ContextAlign with Temporally Aligned Contextual Supervision. FuseCodec establishes new state-of-the-art performance on the LibriSpeech test set (Panayotov et al., 2015) by integrating contextual and semantic guidance into the learning of discrete speech tokens. Specifically, FuseCodec-Fusion achieves the best scores in transcription accuracy (WER 3.99, WIL 6.45), intelligibility (STOI 0.95), and perceptual quality (ViSQOL 3.47, PESQ 3.13), outperforming EnCodec (Défossez et al., 2022), SpeechTokenizer (Zhang et al., 2024), and DM-Codec (Ahasan et al., 2024). FuseCodec-Distill further achieves the highest UTMOS (3.65) and speaker similarity (0.996), highlighting its strength in perceptual naturalness and speaker fidelity. Meanwhile, FuseCodec-ContextAlign provides a strong trade-off between interpretability and performance, with particularly competitive scores in UTMOS (3.65) and similarity (0.995). These results underscore the effectiveness of incorporating contextual and semantic signals into the tokenization process for high-quality speech reconstruction.

Therefore, our key contributions are:

- We introduce three novel neural codecs based on our method: Latent Representation Fusion (FuseCodec-Fusion), Global Semantic-Contextual Supervision (FuseCodec-Distill), and Temporally Aligned Contextual Supervision (FuseCodec-ContextAlign).
- Our framework tackles different limitations of neural codecs by integrating semantic and contextual information through distinct methods, improving cross-modal alignment and enhancing discrete representation learning.
- We demonstrate the utility of our approach in a downstream TTS model and validate each component with extensive ablation studies.
- FuseCodec achieves state-of-the-art performance on LibriSpeech reducing transcription error and improving speech naturalness.

2 Proposed Method

As shown in Figure 1, we first introduce the speech discretization pipeline (§2.1) and describe the extraction of semantic and contextual representations from pre-trained models (§2.2). We then present three strategies for integrating multimodal guidance into speech tokenization: (i) Latent Representation Fusion (§2.3.1), (ii) Global Semantic-Contextual Supervision (§2.3.2), and (iii) Temporally Aligned Contextual Supervision (§2.3.3). Finally, we outline the training objective (§2.4) and the extension to a text-to-speech task (§2.5).

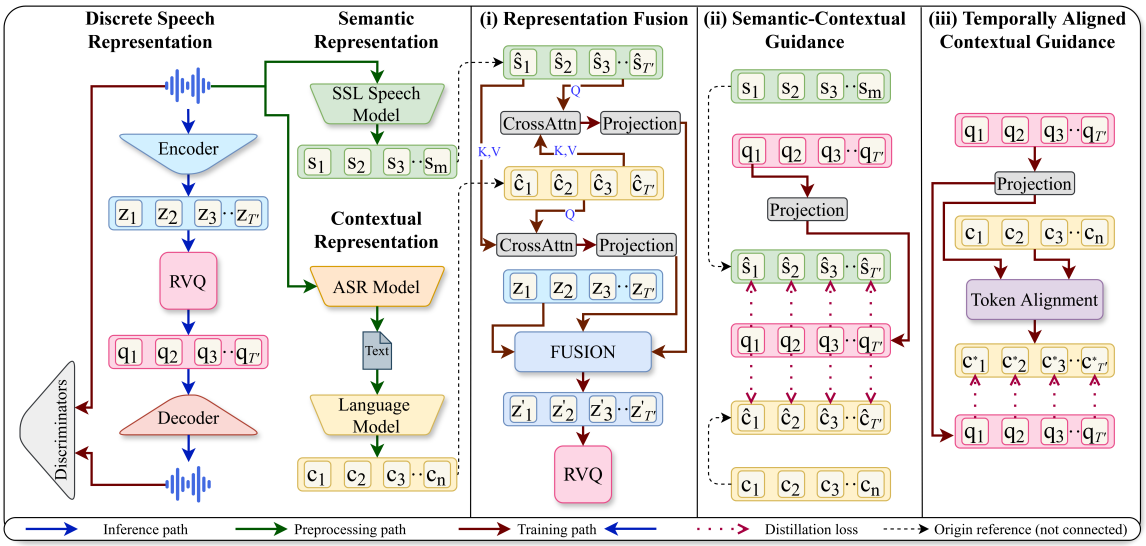

Figure 1: Overview of the FuseCodec speech tokenization framework. Input speech $\mathbf{x}$ is encoded into latent features $\mathbf{Z}$, then quantized into discrete tokens $\mathbf{Q}^{(1:K)}$ via residual vector quantization (RVQ). To enrich these tokens, we incorporate semantic ($\mathbf{S}_i, \hat{\mathbf{S}}$) and contextual ($\mathbf{C}_i, \hat{\mathbf{C}}, \mathbf{C}^*$) representations from frozen pre-trained models. Global vectors $\hat{\mathbf{S}}$ and $\hat{\mathbf{C}}$ are formed via mean pooling and [CLS] selection, respectively. We propose three strategies: (i) Latent Representation Fusion, injecting global vectors $\hat{\mathbf{S}}, \hat{\mathbf{C}}$ with $\mathbf{Z}$ to yield fused latent $\mathbf{Z}'$; (ii) Global Semantic-Contextual Supervision, supervising $\mathbf{Q}^{(1)}$ with global vectors; and (iii) Temporally Aligned Contextual Supervision, aligning full contextual embeddings $\{\mathbf{C}_i\}$ to RVQ outputs via a windowed matching algorithm to form $\mathbf{C}^*$.

2.1 Discrete Speech Representation

Discrete tokens serve as the foundation of neural codec-based speech-language models. Following established approaches (Défossez et al., 2022; Zhang et al., 2024; Ahasan et al., 2024), we discretize audio using an encoder-quantizer setup.

Given an input speech waveform $\mathbf{x}$, an encoder E compresses $\mathbf{x}$ into a sequence of latent representations $\mathbf{Z} = \{\mathbf{z}_i\}_{i=1}^{T'}$, where T' is the number of encoded frames. The encoder output $\mathbf{Z}$ is then passed through a Residual Vector Quantization module (RVQ), consisting of K quantization layers. Each layer k produces a sequence of token indices $\{q_i^{(k)}\}_{i=1}^{T'}$. For each token index $q_i^{(k)}$, we retrieve its corresponding embedding from the k-th codebook, resulting in a sequence of quantized vectors $\mathbf{Q}^{(k)} = \{\mathbf{q}_i^{(k)}\}_{i=1}^{T'}$, where $\mathbf{q}_i^{(k)} \in \mathbb{R}^D$, with D denoting the embedding dimensionality. We use the embeddings from the first quantization layer $\mathbf{Q}^{(1)}$ as the discrete representation of speech, guided with multimodal representations.

2.2 Multimodal Representation Extraction

Concurrently, we extract representations from pre-trained models. Specifically, we obtain contextual representations from a pre-trained language model, which are dynamic, token-level embeddings that adapt to surrounding text (Devlin et al., 2019b; Peters et al., 2018). In parallel, we derive semantic representations from a pre-trained self-supervised speech model, which capture the high-level structure and meaning (Borsos et al., 2023).

Contextual Representation. The input speech waveform $\mathbf{x}$ is transcribed into text $\mathbf{x}'$ using a pre-trained Automatic Speech Recognition (ASR) model A, such that $\mathbf{x}' = A(\mathbf{x})$. The ASR model functions purely as a speech-to-text converter and remains detached during training. The transcribed text $\mathbf{x}'$ is processed by a pre-trained language model B, which produces a token sequence $\{c_i\}_{i=1}^{n}$. For each token c_i, we extract hidden states from all L layers, represented as $\{\mathbf{h}_i^{(l)}\}_{l=1}^{L}$. These are averaged to produce contextual embeddings: $\mathbf{C}_i = \frac{1}{L}\sum_{l=1}^{L} \mathbf{h}_i^{(l)}$, where $\mathbf{C}_i \in \mathbb{R}^{D'}$, and D' denotes the hidden dimension of the language model.

Semantic Representation. The input speech waveform $\mathbf{x}$ is passed through a pre-trained self-supervised speech model H, which outputs a sequence of frame-level tokens $\{s_i\}_{i=1}^{m}$,. For each frame s_i, we extract hidden states from all L layers: $\{\mathbf{h}_i^{(l)}\}_{l=1}^{L}$. These are averaged to obtain semantic embeddings: $\mathbf{S}_i = \frac{1}{L}\sum_{l=1}^{L} \mathbf{h}_i^{(l)}$, where $\mathbf{S}_i \in \mathbb{R}^{D'}$, and D' denotes the hidden dimension.

2.3 Semantic-Contextual Guidance

Our goal is to enrich discrete speech representations by integrating contextual and semantic information, enabling tighter alignment between

acoustic structure and linguistic meaning. Prior work has explored similar directions: Zhang et al. (2024); Défossez et al. (2024) aligned HuBERT-based semantic features with the first RVQ layer using cosine similarity, while Ahasan et al. (2024) matched BERT-based embeddings to RVQ outputs via padded sequences and similarity loss. However, these methods either rely on a single modality (semantic in Zhang et al. (2024); Défossez et al. (2024)) or lack robust cross-modal alignment (misaligned context in Ahasan et al. (2024)).

In contrast, we unify semantic and contextual representations while ensuring robust alignment. For this, we propose three strategies: (i) Latent Representation Fusion (§2.3.1), (ii) Global Semantic-Contextual Supervision (§2.3.2), and (iii) Temporally Aligned Contextual Supervision (§2.3.3)

2.3.1 Latent Representation Fusion

We first propose fusing semantic and contextual representations with the encoder's latent output. The enhanced latents are then passed to the residual vector quantization (RVQ) module, enabling the learning of discrete codes enriched with semantic and contextual information.

We begin by obtaining global semantic and contextual representations. Specifically, we take the average of semantic embeddings $\{\mathbf{S}_i\}_{i=1}^{m}$ to compute the global semantic vector $\hat{\mathbf{S}} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{S}_i$. For the textual modality, we select the `[CLS]` token embedding from the contextual representations $\{\mathbf{C}_i\}_{i=1}^{n}$, yielding $\hat{\mathbf{C}} = \mathbf{C}_{\texttt{[CLS]}}$.

We then broadcast each global vector across the discrete token sequence length T', forming: $\tilde{\mathbf{S}} = \{\hat{\mathbf{S}}\}_{t=1}^{T'}$, and $\tilde{\mathbf{C}} = \{\hat{\mathbf{C}}\}_{t=1}^{T'}$. Broadcasting allows each token to inherit the full semantic or contextual knowledge of the sequence, ensuring every position is enriched with the most informative signal for cross-modal fusion or distillation.

Next, we apply multi-head cross-attention to enable cross-modal interaction, followed by an MLP projection to match the encoder dimension D:

$$\begin{aligned}\mathbf{S}' &= \text{CrossAttention}(\tilde{\mathbf{S}}, \tilde{\mathbf{C}}, \tilde{\mathbf{C}})\mathbf{W}_S, \\ \mathbf{C}' &= \text{CrossAttention}(\tilde{\mathbf{C}}, \tilde{\mathbf{S}}, \tilde{\mathbf{S}})\mathbf{W}_C,\end{aligned} \tag{1}$$

where $\mathbf{W}_S, \mathbf{W}_C \in \mathbb{R}^{D' \times D}$ are learned projection matrices and CrossAttention$(\cdot)$ denotes multi-head cross-attention. Finally, we fuse the modality signals with the latent representation $\mathbf{Z} \in \mathbb{R}^{T' \times D}$ via additive fusion and modality dropout:

$$\mathbf{Z}' = \mathbf{Z} + (\mathbf{S}' \odot \mathcal{D}_S) + (\mathbf{C}' \odot \mathcal{D}_C), \tag{2}$$

where $\mathcal{D}_S, \mathcal{D}_C \in \{0,1\}^{T' \times D}$ are stochastic dropout masks applied during training. Dropout promotes robustness by preventing the quantized representations from over-relying on the fused modalities (Hussen Abdelaziz et al., 2020), and allows inference using only the encoder signal. The resulting fused representation $\mathbf{Z}'$ is then passed to the RVQ module for discrete speech quantization.

2.3.2 Semantic-Contextual Supervision

In addition to latent fusion, we explore an alternative representation supervision strategy, motivated by its effectiveness of similarity matching in prior speech tokenization work (Zhang et al., 2024; Défossez et al., 2024; Ahasan et al., 2024). Unlike previous methods that supervise over feature dimensions or require local frame-level alignment, we introduce a global-to-local time-axis distillation scheme. Specifically, we use global semantic $\hat{\mathbf{S}}$ and contextual $\hat{\mathbf{C}}$ vectors to supervise the RVQ output across time. This provides temporally consistent guidance and encourages the quantized space to capture modality-aware temporal dynamics.

We adapt the *combined distillation loss* from Ahasan et al. (2024), proposing it to operate along the temporal axis rather than the feature axis. This modification enables more effective alignment of discrete latent representations with temporally distributed semantic and contextual signals, enhancing cross-modal coherence over time.

Given the broadcasted global signals (see 2.3.1) $\tilde{\mathbf{S}}, \tilde{\mathbf{C}} \in \mathbb{R}^{T' \times D'}$, we apply a linear projection to the first-layer RVQ output $\mathbf{Q}^{(1)} \in \mathbb{R}^{T' \times D}$ to align dimensionality: $\mathbf{Q}'^{(1)} = \mathbf{Q}^{(1)}\mathbf{W}$, where $\mathbf{W} \in \mathbb{R}^{D \times D'}$. Finally, we apply semantic-contextual supervision using a temporally-aware distillation loss.

$$\begin{aligned}\mathcal{L}_{\text{distill}} = -\frac{1}{T'}\sum_{t=1}^{T'}\log\sigma\left(\frac{1}{2}\left[\cos\left(\mathbf{Q}_t'^{(1)}, \tilde{\mathbf{S}}_t\right)\right.\right. \\ \left.\left. + \cos\left(\mathbf{Q}_t'^{(1)}, \tilde{\mathbf{C}}_t\right)\right]\right),\end{aligned} \tag{3}$$

where $\sigma(\cdot)$ is the sigmoid function and $\cos(\cdot,\cdot)$ denotes cosine similarity. This formulation provides fine-grained temporal supervision using global modality signals, enhancing the representational quality of the learned discrete tokens.

2.3.3 Aligned Contextual Supervision

Building on our use of the global contextual vector $\hat{\mathbf{C}}$ for supervision, we propose a finer-grained approach that leverages the full sequence of contextual embeddings $\{\mathbf{C}_i\}_{i=1}^{n}$ to supervise the RVQ to-

ken sequence $\{\mathbf{q}_t^{(1)}\}_{t=1}^{T'}$, enabling richer, timestep-level guidance. A key challenge, however, is the mismatch in sequence lengths between the contextual embeddings (n) and the RVQ output (T').

To address this, we propose a dynamic window-based alignment strategy that assigns each contextual embedding $\mathbf{C}_i \in \mathbb{R}^{D'}$ to the most similar RVQ token $\mathbf{Q}_t^{(1)}$ embedding within a localized search window using cosine similarity. A dynamic window-shifting mechanism prevents alignment overlap and ensures sequence-wide consistency. If multiple RVQ tokens within the window share the highest similarity, $\mathbf{C}_i$ is assigned to all corresponding positions in the aligned output $\mathbf{C}^*$, accounting for cases where a single linguistic token spans multiple speech frames. The resulting sequence $\mathbf{C}^* \in \mathbb{R}^{T' \times D'}$ enables timestep-level supervision. The full procedure is detailed in Algorithm 1.

Algorithm 1 Window-Based Token Alignment

Require: Contextual embeddings $\{\mathbf{C}_i\}_{i=1}^{n}$, RVQ tokens $\{\mathbf{Q}_t^{(1)}\}_{t=1}^{T'}$, optional window size w

if w not provided **then**
  $w \leftarrow \lfloor T'/n \rfloor$
end if
Initialize aligned output $\mathbf{C}^* \in \mathbb{R}^{T' \times D'} \leftarrow 0$
Initialize $\ell \leftarrow 0$ {last matched index}
for $i = 1$ to n **do**
  if dynamic window **then**
    $s \leftarrow \ell + 1$ if $i > 1$, else 0 {start index}
    $e \leftarrow \min(s + w, T')$ {end index}
  else
    $s \leftarrow (i-1) \cdot w$, $e \leftarrow \min(s + w, T')$
  end if
  Compute cosine similarity $\alpha_t = \cos(\mathbf{C}_i, \mathbf{Q}_t^{(1)})$ for $t \in [s, e)$
  Let $\tau \leftarrow \max_t \alpha_t$ {maximum similarity}
  $\mathcal{T}_i \leftarrow \{t \mid \alpha_t \geq \tau\}$
  for each $t \in \mathcal{T}_i$ **do**
    $\mathbf{C}_t^* \leftarrow \mathbf{C}_i$
  end for
  $\ell \leftarrow \max(\mathcal{T}_i)$
end for
return $\mathbf{C}^*$

Finally, we apply temporally aligned contextual supervision using a timestep-level distillation loss:

$$\mathcal{L}_{\text{distill}} = -\frac{1}{T'} \sum_{t=1}^{T'} \log \sigma \left(\cos \left(\mathbf{Q}_t'^{(1)}, \mathbf{C}_t^* \right) \right), \quad (4)$$

where $\mathbf{Q}'^{(1)} = \mathbf{Q}^{(1)}\mathbf{W} \in \mathbb{R}^{T' \times D'}$ is the linearly projected RVQ output, and $\sigma(\cdot)$ denotes the sigmoid function. This loss enforces temporally precise alignment between acoustic tokens and their corresponding contextual representations, encouraging modality-aware token learning.

2.4 Architecture and Training Objective

We build on widely adopted neural codec architectures and training objectives, following (Défossez et al., 2022; Zhang et al., 2024; Ahasan et al., 2024), to establish a strong and reliable foundation. We contribute to enhancing the learned representations through semantic and contextual supervision and fusion without altering the model architecture.

We use wav2vec 2.0 (base-960h) as the ASR model A (Baevski et al., 2020), BERT (bert-base-uncased) as the language model B (Devlin et al., 2019a), and HuBERT (base-ls960) as the self-supervised speech model H (Hsu et al., 2021). All pre-trained models are frozen during training. The speech tokenizer consists of an encoder E, an RVQ module with 8 quantization layers (codebooks) of size 1024, a decoder D, and three discriminators (multi-period, multi-scale, and multi-scale STFT). Architectural details are provided in Appendix B.1.

Quantization operates on 50 Hz frame rates. The encoder and RVQ use an embedding dimension of $D = 1024$, while the pre-trained langauge and speech model have $D' = 768$. Cross-Attentions are implemented using 8-heads. The dropout masks $\mathcal{D}_S$ and $\mathcal{D}_C$ are applied at a rate of 10%.

Training Objective. We also adopt a multi-objective training setup grounded in established neural codec practices. This includes time-domain reconstruction loss $\mathcal{L}_{\text{time}}$, frequency-domain reconstruction loss $\mathcal{L}_{\text{freq}}$, adversarial loss $\mathcal{L}_{\text{gen}}$, feature matching loss $\mathcal{L}_{\text{feat}}$, and RVQ commitment loss $\mathcal{L}_{\text{commit}}$ (see Appendix B.2 for details).

To further enhance representation quality, we introduce two auxiliary supervision objectives: a global distillation loss as $\mathcal{L}_{\text{distill}}$ (Sec. 2.3.2) and a temporally aligned contextual loss as $\mathcal{L}_{\text{distill}}$ (Sec. 2.3.3). $\mathcal{L}_{\text{distill}}$ is set to 0, when applying the Latent Representation Fusion (Sec. 2.3.1).

The final training objective is a weighted sum:

$$\begin{aligned} \mathcal{L}_{\text{total}} = {} & \lambda_{\text{time}}\mathcal{L}_{\text{time}} + \lambda_{\text{freq}}\mathcal{L}_{\text{freq}} + \lambda_{\text{gen}}\mathcal{L}_{\text{gen}} \\ & + \lambda_{\text{feat}}\mathcal{L}_{\text{feat}} + \lambda_{\text{commit}}\mathcal{L}_{\text{commit}} \\ & + (\lambda_{\text{distill}}\mathcal{L}_{\text{distill}} \text{ or } 0) \end{aligned} \quad (5)$$

2.5 Downstream Extension to TTS Model

We extend the learned discrete token representations to a downstream text-to-speech (TTS) task, following the neural codec language modeling framework and objective used in prior work (Wang et al., 2023; Zhang et al., 2024; Ahasan et al., 2024). In this paradigm, speech synthesis is performed by predicting quantized acoustic tokens produced by the RVQ and decoded by a neural codec.

For this, we propose FuseCodec-TTS, an extension of FuseCodec-Fusion trained with either Latent Representation Fusion (see §2.3.1) or FuseCodec-Distill using Global Semantic-Contextual Supervision (see §2.3.2). This allows the TTS model to operate on discrete speech tokens enriched with semantic and contextual information.

Given a phoneme sequence $\mathbf{p}$ and an acoustic prompt $\mathbf{A} \in \mathbb{R}^{\tau \times K}$ extracted from a reference utterance using FuseCodec, the goal is to predict a sequence of discrete token indices $q^{(1)}, \ldots, q^{(K)}$, corresponding to the K RVQ layers.

To model coarse content and prosodic structure, we autoregressively predict the token indices $q^{(1)}$ from the first quantizer using a decoder-only Transformer conditioned on the phoneme sequence $\mathbf{p}$. The autoregressive (AR) training objective is:

$$\mathcal{L}_{\text{AR}} = -\log \prod_{i=1}^{T'} p\big(q_i^{(1)} \mid q_{<i}^{(1)}, \mathbf{p}; \theta_{\text{AR}}\big) \quad (6)$$

To capture fine-grained acoustic details, we use a non-autoregressive model to predict $q^{(k)}$ for each $k = 2, \ldots, K$, conditioned on the previously predicted layers $q^{(<k)}$, the phoneme sequence $\mathbf{p}$, and the acoustic prompt $\mathbf{A}$. The non-autoregressive (NAR) training objective is:

$$\mathcal{L}_{\text{NAR}} = -\log \prod_{k=2}^{K} p\big(q^{(k)} \mid q^{(<k)}, \mathbf{p}, \mathbf{A}; \theta_{\text{NAR}}\big) \quad (7)$$

Both AR and NAR token generators are implemented using 12-layer Transformers with 16 attention heads, 1024-dimensional embeddings, 4096-dimensional feed-forward layers, and a dropout rate of 0.1. The predicted token indices are mapped to their corresponding quantized embeddings $\mathbf{Q}^{(k)}$, which are then passed to FuseCodec's decoder to reconstruct the synthesized speech waveform.

3 Experimental Setup

Dataset. Following prior work in speech tokenization (Zhang et al., 2024; Ahasan et al., 2024), we train FuseCodec on the LibriSpeech (Panayotov et al., 2015) `train-clean-100` subset, which contains 100 hours of English speech from 251 speakers, sampled at 16 kHz. During training, we randomly crop 3-second audio segments and reserve 100 samples for validation. For FuseCodec-TTS, we combine the train and dev subsets of LibriTTS (Zen et al., 2019), comprising 570 hours of speech.

For evaluating FuseCodec, we use the LibriSpeech `test-clean` subset, which comprises 2,620 utterances held out entirely from training. This setup follows prior baselines (Zhang et al., 2024; Ahasan et al., 2024), though we evaluate on the full set rather than a sampled subset. For FuseCodec-TTS, we adopt two established benchmark protocols. In the LibriSpeech evaluation, following Wang et al. (2023), we select utterances between 4 and 10 seconds, yielding a 2.2-hour subset. For each synthesis, a 3-second enrollment segment is randomly cropped from a different utterance by the same speaker. In the VCTK evaluation, following Zhang et al. (2024), a 3-second prompt is selected or cropped from one utterance, and the transcript of a separate utterance from the same speaker serves as the synthesis target.

Training. FuseCodec is trained for 100 epochs on two A40 GPUs with a batch size of 6, using the Adam optimizer with a learning rate of 1×10^{-4} and exponential decay factor 0.98. FuseCodec-TTS is trained on A100 and L40S GPUs. The AR model is trained for 200 epochs, and the NAR model for 150 epochs. Training employs dynamic batching, with each batch containing up to 550 seconds of audio for AR and 100–200 seconds for NAR. We use the ScaledAdam optimizer with a learning rate of 5×10^{-2} and 200 warm-up steps.

Reproducibility. We provide a fully reproducible setup, including a Dockerized environment, source code, model checkpoints, and configuration files (anonymized for review; see Appendix A.)

Baselines. We compare FuseCodec against both established and recent strong baseline speech tokenizers, including EnCodec (24 kHz) (Défossez et al., 2022) and SpeechTokenizer (Zhang et al., 2024), as well as BigCodec (Xin et al., 2024), DAC (16 kHz) (Kumar et al., 2023), DM-Codec (LM+SM) (Ahasan et al., 2024) FACodec (NaturalSpeech 3) (Ju et al., 2024), and Moshi (Défossez et al., 2024). All baseline results are obtained using official released checkpoints. For FuseCodec-TTS, we compare with neural codec language models that incorporate external representation guidance. Specifically, we compare against USLM

Table 1: Speech reconstruction results across content preservation and naturalness metrics. Orange and light orange cells indicate the best and second-best scores, respectively. Results show that FuseCodec variants outperform baselines by unifying contextual and semantic signals in the discrete speech representations.

Model	Content Preservation			Speech Naturalness			
	WER↓	WIL↓	STOI↑	ViSQOL↑	PESQ↑	UTMOS↑	Similarity↑
BigCodec	4.58	7.45	0.93	3.02	2.68	3.44	0.996
DAC	4.09	6.54	0.94	3.36	2.72	3.33	0.996
DM-Codec	4.09	6.75	0.93	3.20	2.77	3.45	0.994
EnCodec	4.04	6.58	0.92	3.06	2.31	2.41	0.980
FACodec	4.11	6.58	0.95	3.11	2.89	3.45	0.996
Mimi	11.61	18.05	0.85	2.49	1.69	2.28	0.934
SpeechTokenizer	4.16	6.71	0.92	3.08	2.60	3.41	0.996
FuseCodec (Baseline)	4.62	7.44	0.93	2.95	2.54	3.18	0.990
FuseCodec-ContextAlign	4.15	6.70	0.93	3.18	2.85	3.65	0.995
FuseCodec-Distill	4.09	6.60	0.94	3.43	3.06	3.65	0.996
FuseCodec-Fusion	3.99	6.45	0.95	3.47	3.13	3.63	0.995

(from SpeechTokenizer) (Zhang et al., 2024) and DM-Codec-TTS (Ahasan et al., 2024), using their official released LibriTTS trained checkpoints.

Metrics. We evaluate FuseCodec using two complementary categories of metrics: *Content Preservation* and *Speech Naturalness*. To assess Content Preservation, we transcribe generated speech using Whisper (medium) (Radford et al., 2023) and compare it to ground-truth text. We report *Word Error Rate (WER)*, defined as WER $= \frac{S+D+I}{N}$, where S, D, and I denote the number of substitutions, deletions, and insertions, and N is the number of words in the reference. We also report *Word Information Lost (WIL)*, given by WIL $= 1 - \frac{C}{N} + \frac{C}{P}$, where C is the number of correct words, N is the number of words in the reference, and P is the number of words in the prediction. Additionally, we include *Short-Time Objective Intelligibility (STOI)*, a reference-based metric estimating intelligibility via short-time spectral similarity. For Speech Naturalness, we evaluate perceptual and acoustic fidelity using both reference-based and learned metrics. *ViSQOL* and *PESQ* assess perceptual quality by modeling auditory similarity and signal distortion, respectively. We also report *UTMOS* for estimating human-judged naturalness, which is a neural MOS predictor trained on large-scale human ratings. Lastly, we compute *Similarity* as the cosine similarity between L2-normalized speaker embeddings extracted using WavLM-TDNN (Chen et al., 2022), reflecting speaker or content consistency. For FuseCodec-TTS, we omit metrics requiring reference audio (e.g., STOI, ViSQOL, PESQ), as exact references are unavailable in synthesis.

4 Experimental Results and Discussion

In this section, we evaluate our proposed methods on speech reconstruction quality (§4.1), their extension to speech synthesis (§4.2), and validate the contribution of each component through comprehensive ablation studies (Appendix C)).

4.1 Speech Reconstruction Evaluation

We evaluate our three proposed methods: (i) Latent Representation Fusion: FuseCodec-Fusion (Sec. 2.3.1) (ii) Global Semantic-Contextual Supervision: FuseCodec-Distill (Sec. 2.3.2), and (iii) Temporally Aligned Contextual Supervision: FuseCodec-ContextAlign (Sec. 2.3.3).

Results. The results in Table 1 show that FuseCodec improves performance across all metrics related to content preservation and speech naturalness. FuseCodec-Fusion performs best overall, achieving the lowest WER (3.99) and WIL (6.45), along with the highest STOI (0.95), reducing transcription error and improving intelligibility. It also achieves the highest scores in ViSQOL (3.47) and PESQ (3.13), reflecting superior perceptual quality. FuseCodec-Distill attains top scores in UTMOS (3.65) and Similarity (0.996), while also ranking second in STOI (0.94), ViSQOL (3.43), and PESQ (3.06), demonstrating strong naturalness and speaker consistency. FuseCodec-ContextAlign also performs competitively, particularly in UTMOS (3.65) and Similarity (0.995), while showing consistent improvements over FuseCodec (Baseline).

Discussion. FuseCodec-Fusion achieves the best overall performance. Compared to EnCodec, which focuses purely on acoustic representations, and FACodec, which separates attribute learning without unifying representations, FuseCodec-

Table 2: Zero-shot TTS evaluation on LibriSpeech and VCTK. FuseCodec-TTS variants are compared to official neural codec-based TTS checkpoints trained on LibriTTS. Bold and underline indicate best and second-best scores. FuseCodec-TTS improves intelligibility, similarity, and naturalness via semantic-contextual aware tokenization.

Model	WER ↓		WIL ↓		Similarity ↑		UTMOS ↑	
	LibriSpeech	VCTK	LibriSpeech	VCTK	LibriSpeech	VCTK	LibriSpeech	VCTK
DM-Codec-TTS	10.26	5.02	13.79	8.21	<u>0.82</u>	**0.79**	**3.70**	**3.86**
USLM	16.72	14.79	25.65	23.24	0.80	<u>0.78</u>	2.93	3.01
FuseCodec-Distill-TTS	**8.55**	**3.66**	**12.07**	**6.02**	<u>0.82</u>	<u>0.78</u>	3.55	3.75
FuseCodec-Fusion-TTS	<u>9.67</u>	<u>4.07</u>	<u>13.23</u>	<u>7.18</u>	**0.83**	**0.79**	<u>3.63</u>	<u>3.82</u>

Fusion incorporates both semantic and contextual signals directly into the encoder's latent space. This enables the quantizer to learn a unified representation aligned with both linguistic meaning and acoustic structure. It also outperforms models like DAC and BigCodec, which prioritize compression but lack representational alignment. FuseCodec-Distill improves upon SpeechTokenizer and Mimi, which distill only semantic representations from speech models and underperform on intelligibility and quality. In contrast, FuseCodec-Distill supervises the quantized space with global contextual and semantic signals, promoting alignment with high-level linguistic and acoustic content.

FuseCodec-ContextAlign introduces fine-grained supervision by aligning discrete tokens with temporally matched contextual tokens, encouraging each token to reflect local linguistic context. Although its constrained alignment limits global contextual guidance, leading to slightly lower performance than FuseCodec-Fusion and FuseCodec-Distill, it still outperforms DM-Codec, improving intelligibility and speaker similarity. Overall, contextual guidance is most effective for content preservation, while semantic supervision enhances speech naturalness. FuseCodec-Fusion delivers the best balance, FuseCodec-Distill excels in speaker fidelity, and FuseCodec-ContextAlign offers interpretable gains. These results underscore the benefit of unifying multimodal representations.

4.2 Speech Synthesis Evaluation

We extend our methods to the zero-shot text-to-speech task and evaluate against neural codec TTS models that incorporate representational supervision for comparison. We deicde to adapt FuseCodec-Fusion to FuseCodec-Fusion-TTS and FuseCodec-Distill to FuseCodec-Distill-TTS, based on their superior performance.

Results. Table 2 shows that both FuseCodec-TTS variants outperform prior methods across most metrics on LibriSpeech and VCTK. FuseCodec-Distill-TTS achieves the best content preservation, with the lowest WER (8.55 / 3.66) and WIL (12.07 / 6.02), surpassing both DM-Codec-TTS and USLM. FuseCodec-Fusion-TTS delivers the highest perceptual quality, achieving the top speaker similarity (0.83 / 0.79), while also maintaining strong intelligibility with the second-best UTMOS (3.63 / 3.82), WER (9.67 / 4.07), and WIL (13.23 / 7.18).

Discussion. FuseCodec-Fusion-TTS leads in perceptual quality and speaker similarity. Unlike DM-Codec-TTS, which lacks precise alignment, and USLM, which incorporates only semantic features, FuseCodec-Fusion-TTS integrates both semantic and contextual signals directly into the encoder's latent space. This allows the quantizer to capture expressive prosody and speaker identity, resulting in more natural and coherent speech. FuseCodec-Distill-TTS achieves the highest intelligibility and transcription accuracy. In contrast to USLM's lack of contextual grounding and DM-Codec-TTS's limited supervision, it distills global semantic-contextual representations into the quantized token space, enhancing alignment with semantic and contextual info. While FuseCodec-Fusion-TTS excels in naturalness and speaker fidelity, FuseCodec-Distill-TTS offers stronger linguistic precision. This trade-off reflects the complementary strengths of each variant and underscores the importance of integrating semantic-contextual fusion or supervision into speech tokenization.

5 Conclusion

We introduced FuseCodec, a unified speech tokenization framework that integrates acoustic, semantic, and contextual signals via multimodal representation fusion and supervision. Our methods enable fine-grained alignment and achieve state-of-the-art results on speech reconstruction, improving intelligibility, quality, and speaker similarity. These findings highlight the value of semantic and contextual grounding in discrete speech modeling.

Limitations

While FuseCodec demonstrates strong performance in incorporating semantic and contextual representation into discrete speech tokenization, our approach relies on frozen pretrained language and speech models, which may limit adaptability to non-English or low-resource languages. Moreover, although we focus on zero inference overhead, training requires additional computational resources due to multimodal guidance. Future work could explore lightweight, adaptive guidance mechanisms and broader language coverage.

References

Md Mubtasim Ahasan, Md Fahim, Tasnim Mohiuddin, A K M Mahbubur Rahman, Aman Chadha, Tariq Iqbal, M Ashraful Amin, Md Mofijul Islam, and Amin Ahsan Ali. 2024. Dm-codec: Distilling multimodal representations for speech tokenization. *Preprint*, arXiv:2410.15017.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Preprint*, arXiv:2006.11477.

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. 2013. Estimating or propagating gradients through stochastic neurons for conditional computation. *Preprint*, arXiv:1308.3432.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, and Neil Zeghidour. 2023. Audiolm: A language modeling approach to audio generation. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 31:2523–2533.

Annemarie C. Brown, Eva Childers, Elijah F. W. Bowen, Gabriel A. Zuckerberg, and Richard Granger. 2022. Phonemes in continuous speech are better recognized in context than in isolation. *Frontiers in Communication*, Volume 7 - 2022.

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. 2022. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.

Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. 2016. Fast and accurate deep network learning by exponential linear units (elus). *Preprint*, arXiv:1511.07289.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019a. Bert: Pre-training of deep bidirectional transformers for language understanding. *Preprint*, arXiv:1810.04805.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019b. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. 2022. High fidelity neural audio compression. *Preprint*, arXiv:2210.13438.

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

Mark Hallap, Emmanuel Dupoux, and Ewan Dunbar. 2023. Evaluating context-invariance in unsupervised speech representations. *Preprint*, arXiv:2210.15775.

Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM transactions on audio, speech, and language processing*, 29:3451–3460.

Ahmed Hussen Abdelaziz, Barry-John Theobald, Paul Dixon, Reinhard Knothe, Nicholas Apostoloff, and Sachin Kajareker. 2020. Modality dropout for improved performance-driven talking faces. In *Proceedings of the 2020 International Conference on Multimodal Interaction*, ICMI '20, page 378–386, New York, NY, USA. Association for Computing Machinery.

Zeqian Ju, Yuancheng Wang, Kai Shen, Xu Tan, Detai Xin, Dongchao Yang, Yanqing Liu, Yichong Leng, Kaitao Song, Siliang Tang, Zhizheng Wu, Tao Qin, Xiang-Yang Li, Wei Ye, Shikun Zhang, Jiang Bian, Lei He, Jinyu Li, and Sheng Zhao. 2024. Naturalspeech 3: Zero-shot speech synthesis with factorized codec and diffusion models. *Preprint*, arXiv:2403.03100.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. Hifi-gan: generative adversarial networks for efficient and high fidelity speech synthesis. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brebisson, Yoshua Bengio, and Aaron Courville. 2019. Melgan: Generative adversarial networks for conditional waveform synthesis. *Preprint*, arXiv:1910.06711.

Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. 2023. High-fidelity audio compression with improved rvqgan. *Preprint*, arXiv:2306.06546.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine Mcleavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 28492–28518. PMLR.

Craig W Schmidt, Varshini Reddy, Haoran Zhang, Alec Alameddine, Omri Uzan, Yuval Pinter, and Chris Tanner. 2024. Tokenization is more than compression. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 678–702, Miami, Florida, USA. Association for Computational Linguistics.

Arnon Turetzky and Yossi Adi. 2024. Last: Language model aware speech tokenization. *Preprint*, arXiv:2409.03701.

Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. 2018. Neural discrete representation learning. *Preprint*, arXiv:1711.00937.

Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, Lei He, Sheng Zhao, and Furu Wei. 2023. Neural codec language models are zero-shot text to speech synthesizers. *Preprint*, arXiv:2301.02111.

Detai Xin, Xu Tan, Shinnosuke Takamichi, and Hiroshi Saruwatari. 2024. Bigcodec: Pushing the limits of low-bitrate neural speech codec. *Preprint*, arXiv:2409.05377.

Dongchao Yang, Songxiang Liu, Rongjie Huang, Jinchuan Tian, Chao Weng, and Yuexian Zou. 2023. Hifi-codec: Group-residual vector quantization for high fidelity audio codec. *Preprint*, arXiv:2305.02765.

Zhen Ye, Peiwen Sun, Jiahe Lei, Hongzhan Lin, Xu Tan, Zheqi Dai, Qiuqiang Kong, Jianyi Chen, Jiahao Pan, Qifeng Liu, Yike Guo, and Wei Xue. 2024. Codec does matter: Exploring the semantic shortcoming of codec for audio language model. *Preprint*, arXiv:2408.17175.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. 2022. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507.

Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J. Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. 2019. Libritts: A corpus derived from librispeech for text-to-speech. *Preprint*, arXiv:1904.02882.

Xin Zhang, Dong Zhang, Shimin Li, Yaqian Zhou, and Xipeng Qiu. 2024. Speechtokenizer: Unified speech tokenizer for speech language models. In *The Twelfth International Conference on Learning Representations*.

A Resources

We provide all necessary resources to ensure full reproducibility of our models and results. All links are anonymized for double-blind peer review.

- **Docker:** A containerized environment with all required Python packages for training. LINK
- **Code and Configuration:** Full codebase for preprocessing, training, and inference. LINK
- **Model Checkpoints:** Trained model weights. LINK

B Tokenizer Design and Loss Functions

B.1 Model Details

To implement a strong speech tokenizer baseline, we adopt a standard neural codec architecture and discriminator setup commonly used in prior work (Défossez et al., 2022; Zeghidour et al., 2022).

Encoder and Decoder. The Encoder consists of an initial 1D convolutional layer with 32 channels and a kernel size of 7, followed by 4 stacked residual blocks. Each block includes two dilated convolutions with a (3, 1) kernel and no dilation expansion (dilation = 1), a residual connection, and a strided convolutional layer for temporal downsampling. Stride values across the blocks are set to 2, 4, 5, and 8, with kernel sizes for the downsampling layers set to twice the corresponding stride. Channel dimensions double at each downsampling stage. The encoder then includes a two-layer BiLSTM, and concludes with a 1D convolution (kernel size 7) to project to the target embedding dimension. ELU (Clevert et al., 2016) is used as the activation function, and layer normalization or weight normalization is applied depending on the layer. The Decoder mirrors the encoder architecture, with the only difference being the use of transposed convolutions in place of strided convolutions to reverse the downsampling steps, and the inclusion of LSTM layers to restore temporal resolution.

Residual Vector Quantizer. The Residual Vector Quantizer (RVQ) module discretizes the encoder's continuous latent representations into a sequence of codebook indices. Specifically, we quantize the encoder latent tensor of shape $[B, D, T]$ using 8 residual codebooks, each with 1024 codebook entries. Each subsequent codebook quantizes the residual error of the previous one. Codebook entries are updated using an exponential moving average with a decay factor of 0.99. To prevent codebook collapse, unused entries are randomly resampled using vectors from the current batch. The RVQ output is a discrete tensor of shape $[B, N_q, T]$, where N_q is the number of active quantizers. The indices are mapped back to the original latent space by summing the corresponding codebook embeddings and are then fed into the decoder to reconstruct the input. A straight-through estimator (Bengio et al., 2013) is used to propagate gradients through the quantizer.

Discriminators. We utilize discriminators to guide the generators (Encoder, RVQ, and Decoder) to reconstruct speech more closely to the original. We make use of three distinct discriminators: a Multi-Scale STFT (MS-STFT) discriminator, a Multi-Scale Discriminator (MSD), and a Multi-Period Discriminator (MPD). The MS-STFT discriminator, proposed by (Défossez et al., 2022), works on multiple resolutions of the complex-valued short-time Fourier transform (STFT). It treats the real and imaginary parts as concatenated and applies a sequence of 2D convolutional layers. The initial layer uses a kernel size of 3×8 with 32 channels. This is followed by convolutions with increasing temporal dilation rates (1, 2, and 4) and a stride of 2 along the frequency axis. A final 3×3 convolution with stride 1 outputs the discriminator prediction. The MSD processes the raw waveform at various temporal scales using progressively downsampled versions of the input. We adopt the configuration from (Zeghidour et al., 2022), which was originally based on (Kumar et al., 2019). Similarly, the MPD, introduced by (Kong et al., 2020), models periodic structure in the waveform by reshaping it into a 2D input with unique periodic patterns. For consistency, we standardize the number of channels in both the MSD and MPD to match those in the MS-STFT discriminator.

B.2 Training Objective

To ensure that FuseCodec learns discrete speech representations, we ground our training objective on proven techniques, following (Défossez et al., 2022; Zhang et al., 2024; Ahasan et al., 2024).

Reconstruction loss. Let $\mathbf{x}$ and $\hat{\mathbf{x}}$ denote the original and reconstructed speech waveforms, respectively. For spectral comparisons, we define 64-bin Mel-spectrograms $\mathbf{M}_i(\cdot)$ using STFTs with window size 2^i and hop size $2^i/4$, where $i \in \mathcal{E} = \{5, \ldots, 11\}$ indexes different resolution scales. We compute the time-domain $\mathcal{L}_{\text{time}}$ and frequency-

domain $\mathcal{L}_{\text{freq}}$ reconstruction losses as:

$$\mathcal{L}_{\text{time}} = \|\mathbf{x} - \hat{\mathbf{x}}\|_1 \tag{8}$$

$$\mathcal{L}_{\text{freq}} = \sum_{i \in \mathcal{E}} \Big(\|\mathbf{M}_i(\mathbf{x}) - \mathbf{M}_i(\hat{\mathbf{x}})\|_1 + \|\mathbf{M}_i(\mathbf{x}) - \mathbf{M}_i(\hat{\mathbf{x}})\|_2 \Big) \tag{9}$$

Adversarial loss. To reduce the discriminability of reconstructed speech, we adopt a GAN-based training objective with a set of discriminators $\{D^{(i)}\}_{i=1}^{d}$, including multi-period (MPD), multi-scale (MSD), and multi-scale STFT (MS-STFT) variants (see Appendix B for details). The generator $\mathcal{L}_{\text{gen}}$ and discriminator $\mathcal{L}_{\text{disc}}$ losses are computed as:

$$\mathcal{L}_{\text{gen}} = \frac{1}{d} \sum_{i=1}^{d} \max\Big(0,\ 1 - D^{(i)}(\hat{\mathbf{x}})\Big) \tag{10}$$

$$\mathcal{L}_{\text{disc}} = \frac{1}{d} \sum_{i=1}^{d} \big[\max(0,\ 1 - D^{(i)}(\mathbf{x})) + \max(0,\ 1 + D^{(i)}(\hat{\mathbf{x}}))\big] \tag{11}$$

Let $D_j^{(i)}(\cdot)$ denote the output of the j-th layer of $D^{(i)}$, with ℓ total layers. We include a feature $\mathcal{L}_{\text{feat}}$ matching loss to stabilize training and align intermediate features as:

$$\mathcal{L}_{\text{feat}} = \frac{1}{d\ell} \sum_{i=1}^{d} \sum_{j=1}^{\ell} \frac{\|D_j^{(i)}(\mathbf{x}) - D_j^{(i)}(\hat{\mathbf{x}})\|_1}{\text{mean}\Big(\|D_j^{(i)}(\mathbf{x})\|_1\Big)} \tag{12}$$

Commitment Loss. To ensure encoder outputs align closely with their quantized representations, we apply a commitment penalty during residual vector quantization (RVQ). Let $\mathbf{r}_j$ denote the residual vector at step $j \in \{1, \ldots, q\}$, and $\mathbf{c}_j$ be its corresponding nearest codebook entry, we calculate commitment loss $\mathcal{L}_{\text{commit}}$ as:

$$\mathcal{L}_{\text{commit}} = \sum_{j=1}^{q} \|\mathbf{r}_j - \mathbf{c}_j\|_2^2 \tag{13}$$

C Ablation Studies

We ablate and investigate each design choice and the necessity of components in our proposed methodology for FuseCodec. All model hyperparameters, training procedures, and configurations are kept fixed, except for the specific changes introduced in each ablation setup.

C.1 Ablation: Attention-Projection Configuration in Representation Fusion

Setup. We investigate the impact of changing the attention-projection configuration in FuseCodec-Fusion (Section 2.3.1). The selected method, **Cross-Before**, applies multi-head cross-attention prior to projection:

$$\begin{aligned} \mathbf{S}' &= \text{CrossAttention}(\tilde{\mathbf{S}}, \tilde{\mathbf{C}}, \tilde{\mathbf{C}})\mathbf{W}_S, \\ \mathbf{C}' &= \text{CrossAttention}(\tilde{\mathbf{C}}, \tilde{\mathbf{S}}, \tilde{\mathbf{S}})\mathbf{W}_C, \end{aligned} \tag{14}$$

where $\tilde{\mathbf{S}}, \tilde{\mathbf{C}} \in \mathbb{R}^{T' \times D'}$ are broadcasted global semantic and contextual vectors. We compare this with the following ablated variants:

None, which skips attention and directly applies projection:

$$\begin{aligned} \mathbf{S}' &= \tilde{\mathbf{S}}\mathbf{W}_S, \\ \mathbf{C}' &= \tilde{\mathbf{C}}\mathbf{W}_C \end{aligned} \tag{15}$$

Self-Before, which applies self-attention before projection:

$$\begin{aligned} \mathbf{S}' &= \text{SelfAttention}(\tilde{\mathbf{S}}, \tilde{\mathbf{S}}, \tilde{\mathbf{S}})\mathbf{W}_S, \\ \mathbf{C}' &= \text{SelfAttention}(\tilde{\mathbf{C}}, \tilde{\mathbf{C}}, \tilde{\mathbf{C}})\mathbf{W}_C \end{aligned} \tag{16}$$

Self-After, which projects first and then applies self-attention:

$$\begin{aligned} \mathbf{S}' &= \text{SelfAttention}(\tilde{\mathbf{S}}\mathbf{W}_S), \\ \mathbf{C}' &= \text{SelfAttention}(\tilde{\mathbf{C}}\mathbf{W}_C) \end{aligned} \tag{17}$$

Cross-After, which applies projection before cross-attention:

$$\begin{aligned} \mathbf{S}' &= \text{CrossAttention}(\tilde{\mathbf{S}}\mathbf{W}_S, \tilde{\mathbf{C}}\mathbf{W}_C, \tilde{\mathbf{C}}\mathbf{W}_C), \\ \mathbf{C}' &= \text{CrossAttention}(\tilde{\mathbf{C}}\mathbf{W}_C, \tilde{\mathbf{S}}\mathbf{W}_S, \tilde{\mathbf{S}}\mathbf{W}_S) \end{aligned} \tag{18}$$

Results. Table 3 shows the results of five variants. The selected **Cross-Before** setup achieves the highest performance on intelligibility STOI (0.95), and all naturalness metrics: ViSQOL (3.47), PESQ (3.13), and second-best UTMOS (3.63). **Self-Before** yields the best WER (3.92) and WIL (6.36), and second-best ViSQOL (3.43), PESQ (3.05), and

Table 3: Ablation of attention-projection configurations in multimodal latent fusion. **Cross** variants incorporate cross-modal attention between semantic and contextual signals, while **Self** variants apply self-attention. **Before** applies attention prior to projection into the encoder's latent space, whereas **After** applies attention post-projection. **None** uses direct projection without attention. *Applying cross-modal attention before projection consistently improves content preservation and speech naturalness by enabling richer multimodal interactions in the original dimension.*

Model Variant	Attn-Proj Type	Content Preservation			Speech Naturalness			
		WER↓	WIL↓	STOI↑	ViSQOL↑	PESQ↑	UTMOS↑	Similarity↑
FuseCodec-Fusion	None	4.10	6.60	0.93	3.26	2.92	**3.65**	0.995
FuseCodec-Fusion	Self-After	4.07	6.61	0.93	3.26	2.95	3.63	0.995
FuseCodec-Fusion	Self-Before	**3.92**	**6.36**	0.94	3.43	3.05	3.59	0.995
FuseCodec-Fusion	Cross-After	4.17	6.70	0.93	3.28	2.90	3.61	0.995
FuseCodec-Fusion	Cross-Before	3.99	6.45	**0.95**	**3.47**	**3.13**	3.63	0.995

STOI (0.94). The **None** and **Cross-After** configurations perform comparatively worse across intelligibility and naturalness.

Discussion. These results demonstrate that the configuration of attention relative to projection significantly impacts the effectiveness of representation fusion. The best-performing method, **Cross-Before**, applies cross-modal attention in the original lower-dimensional space. This enables richer semantic-contextual interactions to be captured before transformation into the higher-dimensional encoder space, leading to improved intelligibility and perceptual quality.

Self-Before performs competitively by achieving the best WER and WIL, suggesting that intra-modal structuring of global feature representations also benefits the fusion approach. However, the absence of explicit cross-modal exchange limits its effectiveness on naturalness metrics such as UTMOS and PESQ.

By contrast, **Cross-After** performs poorly, indicating that applying cross-attention after projection diminishes its effectiveness. Suggesting that once projected into the higher-dimensional space, the global vectors lose semantic coherence, resulting in less expressive fusion and lower audio quality.

Finally, removing attention (**None**) results in the weakest performance on intelligibility and perceptual scores, despite yielding the highest UTMOS. This indicates that even unstructured modality signals can enhance naturalness, but without alignment through attention mechanisms, they fail to deliver consistent semantic-contextual grounding.

Overall, these results confirm that performing attention prior to projection, especially cross-modal attention, is essential for extracting the most benefit from semantic-contextual signals during fusion.

C.2 Ablation: Attention-Guidance Configuration in Semantic-Contextual Guidance

Setup. We study the impact of attention configuration and guidance modality used in the distillation objective. Our method, FuseCodec-Distill, introduces timestep-aligned supervision using global contextual and semantic signals (Section 2.3.2). The selected configuration, **None + Semantic-Contextual**, projects the first-layer RVQ tokens $\mathbf{Q}^{(1)}$ and computes cosine similarity with both semantic and contextual guidance vectors:

$$\mathcal{L}_{\text{distill}} = -\frac{1}{T'}\sum_{t=1}^{T'} \log \sigma \left(\frac{1}{2} \left[\cos\left(\mathbf{Q}_t'^{(1)}, \tilde{\mathbf{S}}_t \right) + \cos\left(\mathbf{Q}_t'^{(1)}, \tilde{\mathbf{C}}_t \right) \right] \right) \tag{19}$$

We compare this against three ablated variants:

None + Contextual, which excludes both attention and semantic guidance:

$$\mathcal{L}_{\text{distill}} = -\frac{1}{T'}\sum_{t=1}^{T'} \log \sigma \left(\cos\left(\mathbf{Q}_t'^{(1)}, \tilde{\mathbf{C}}_t \right) \right) \tag{20}$$

Cross + Contextual, which introduces cross-attention between contextual vectors and projected RVQ tokens:

$$\tilde{\mathbf{C}} = \text{CrossAttention}(\tilde{\mathbf{C}}, \mathbf{Q}'^{(1)}, \mathbf{Q}'^{(1)}) \tag{21}$$

Cross + Semantic-Contextual, which includes cross-attention but retains both guidance signals.

Results. Table 4 reports the performance across four configurations. The best-performing variant is **None + Semantic-Contextual**, achieving the lowest WER (4.09) and WIL (6.60), and highest

Table 4: Ablation of attention and guidance strategies in semantic-contextual distillation. **Cross** variants apply cross-attention between contextual embeddings and discrete tokens, while **None** applies supervision directly. **Semantic-Contextual** combines both global semantic and contextual signals. *Direct supervision using both signals achieves the best intelligibility and perceptual quality by preserving global structure.*

Model Variant	Attention	Guidance	Content Preservation			Speech Naturalness			
			WER↓	WIL↓	STOI↑	ViSQOL↑	PESQ↑	UTMOS↑	Similarity↑
FuseCodec-Distill	None	Contextual	4.20	6.77	0.93	3.13	2.74	3.60	0.995
FuseCodec-Distill	Cross	Contextual	4.18	6.75	0.93	3.21	2.83	3.60	0.995
FuseCodec-Distill	None	Semantic-Contextual	**4.09**	**6.60**	**0.94**	**3.43**	**3.06**	**3.65**	**0.996**
FuseCodec-Distill	Cross	Semantic-Contextual	4.21	6.82	0.93	3.18	2.84	3.62	0.994

scores on STOI (0.940), ViSQOL (3.43), PESQ (3.06), UTMOS (3.65), and Similarity (0.996). The second-best results are obtained by **Cross + Contextual**, but excluding semantic guidance or using attention degrades performance across all metrics.

Discussion. These results show that including both semantic and contextual supervision is essential for improving the quantization quality of the discrete tokens. The **None + Semantic-Contextual** configuration outperforms all others, highlighting that cosine-based alignment with both modalities provides the most stable and effective guidance during quantized representation learning.

Introducing cross-attention (**Cross**) reduces performance, suggesting that attention distorts the global nature of the guidance signals and makes supervision less consistent across time. The **Cross + Semantic-Contextual** variant also underperforms, despite having access to both guidance sources, indicating that attention interferes with their inherent structure and alignment function.

The **Contextual-only** variants perform comparatively worse, confirming that semantic signals play an important role in guiding the learned representations toward higher-level content fidelity and improved intelligibility.

Overall, these findings support using both guidance signals in their original global forms and applying them directly, without attention, to ensure stable, timestep-aligned distillation.

C.3 Ablation: Fixed vs. Dynamic Window Configuration in Temporal Alignment

Setup. We investigate the effect of **fixed** versus **dynamic** windowing in the token alignment algorithm (Algorithm 1). Our full method, FuseCodec-ContextAlign, aligns each contextual embedding $\mathbf{C}_i \in \mathbb{R}^{D'}$ to a localized region of RVQ tokens $\{\mathbf{Q}_t^{(1)}\}_{t=1}^{T'}$ based on cosine similarity. The selected configuration, **Dynamic-window Contextual** (see Section 2.3.3), dynamically adjusts the alignment window for each $\mathbf{C}_i$, using the index of the previous match to guide the next search range. This content-aware strategy produces a temporally aligned sequence $\mathbf{C}^* \in \mathbb{R}^{T' \times D'}$, which is used to compute a timestep-level distillation loss:

$$\mathcal{L}_{\text{align}} = -\frac{1}{T'} \sum_{t=1}^{T'} \log \sigma \left(\cos \left(\mathbf{Q}_t'^{(1)}, \mathbf{C}_t^* \right) \right) \quad (22)$$

We compare this setup against the following ablated variants:

Fixed-window Contextual, which uses a fixed alignment window of size $w = \lfloor T'/n \rfloor$, where T' is the RVQ sequence length and n is the number of contextual embeddings. Each $\mathbf{C}_i$ is aligned to the most similar token $\mathbf{Q}_t^{(1)}$ within its predefined window.

Fixed-window Semantic-Contextual, which adds semantic supervision using semantic representations $\{\mathbf{S}_i\}_{i=1}^m$, in addition to contextual representations aligned via a fixed-window token alignment. Since both semantic and RVQ tokens are extracted at the same frame rate, they are inherently time-aligned, requiring no additional alignment. The combined loss is:

$$\mathcal{L}_{\text{align}} = -\frac{1}{T'} \sum_{t=1}^{T'} \log \sigma \left(\frac{1}{2} \left[\cos \left(\mathbf{Q}_t'^{(1)}, \mathbf{C}_t^* \right) + \cos \left(\mathbf{Q}_t'^{(1)}, \mathbf{S}_t \right) \right] \right) \quad (23)$$

Dynamic-window Semantic-Contextual, which replaces the fixed window with a dynamic alignment strategy, while also incorporating direct supervision from semantic embeddings $\{\mathbf{S}_t\}$.

Results. As shown in Table 5, the **Dynamic-window Contextual** configuration achieves the

Table 5: Ablation of windowing and guidance strategies in temporally aligned contextual supervision. **Dynamic** variants adapt the alignment window per token based on content similarity, while **Fixed** variants use a uniform window. **Semantic-Contextual** combines semantic and contextual signals for supervision. *Dynamic windowing consistently improves intelligibility and clarity by enabling finer temporal alignment of contextual embeddings.*

Model Variant	Window	Guidance	Content Preservation			Speech Naturalness			
			WER↓	WIL↓	STOI↑	ViSQOL↑	PESQ↑	UTMOS↑	Similarity↑
FuseCodec-ContextAlign	Fixed	Contextual	4.26	6.88	0.92	**3.19**	2.71	3.58	0.994
FuseCodec-ContextAlign	Dynamic	Contextual	**4.15**	**6.70**	**0.93**	3.18	**2.85**	3.65	0.995
FuseCodec-ContextAlign	Fixed	Semantic-Contextual	4.30	6.88	0.92	3.10	2.62	3.74	0.995
FuseCodec-ContextAlign	Dynamic	Semantic-Contextual	4.21	6.78	0.93	3.12	2.72	**3.75**	0.995

best performance across content preservation metrics, achieving the lowest WER (4.15), WIL (6.70), and highest STOI (0.93). It also performs strongly in terms of speech naturalness, with the best PESQ (2.85), high ViSQOL (3.18), and top Similarity (0.995). The **Dynamic Semantic-Contextual** variant achieves the best UTMOS (3.75), second-best WER (4.21) and WIL (6.78), and matches the top Similarity. By contrast, both **Fixed-window** configurations obtains lower scores across most metrics, particularly the **Fixed Semantic-Contextual** configuration, which scores the lowest ViSQOL (3.10) and PESQ (2.62), despite a relatively high UTMOS (3.74).

Discussion. These results highlight the importance of the temporal alignment strategy in influencing speech reconstruction quality. The superior performance of the **Dynamic-window Contextual** variant demonstrates that token alignment using a dynamic window, where contextual embeddings are adaptively aligned based on token similarity, achieves better semantic grounding and contextual precision.

In contrast, the **Fixed-window** variants suffer from rigid alignment constraints. They fail to capture fine-grained temporal dependencies by enforcing a fixed windowing strategy, which resuls in degraded speech clarity (lower ViSQOL and PESQ). This limitation is especially noticeable in the **Fixed Semantic-Contextual** setup, where the addition of semantic supervision is insufficient to compensate for the strictly aligned contextual embeddings as the fixed window does not account for local content variations.

Both **Semantic-Contextual** variants improve UTMOS, indicating that semantic supervision contributes positively to speech naturalness. However, this comes with a trade-off when not paired with dynamically aligned contextual guidance, as the semantic-only supervision fails to improve content accuracy.

Overall, these findings underscore that dynamic alignment is essential for effective contextual representation guidance. They also highlight that while semantic supervision enhances fluency and naturalness, it must be combined with flexible alignment mechanisms to avoid compromising content preservation.

C.4 Ablation: Dropout Mask Configuration in Representation Fusion

Setup. We investigate the effect of modality dropout rate on the quality of latent representation fusion. As described in Section 2.3.1, we apply stochastic dropout masks $\mathcal{D}_S, \mathcal{D}_C \in \{0,1\}^{T' \times D}$ element-wise to the projected semantic ($\mathbf{S}'$) and contextual ($\mathbf{C}'$) vectors during training:

$$\mathbf{Z}' = \mathbf{Z} + (\mathbf{S}' \odot \mathcal{D}_S) + (\mathbf{C}' \odot \mathcal{D}_C) \tag{24}$$

This stochastic masking prevents FuseCodec from over-reliance on any single modality and encourages the model to learn robust representations.

The selected configuration uses a **10%** dropout rate—i.e., each element in $\mathcal{D}_S$ and $\mathcal{D}_C$ has a 10% chance of being masked to zero during training. We compare this against higher dropout rates: **30%**, **50%**, **70%**, and **90%**.

Results. The best overall performance is achieved with the 10% dropout rate configuration, which achieves the lowest WER (3.99) and WIL (6.45) and the highest STOI (0.95), ViSQOL (3.47), and PESQ (3.13). Increasing the dropout rate to 30–90% leads to the worsening of the most content preservation and speech naturalness metrics. While UTMOS and Similarity remain relatively stable, 50% dropout achieves minor gains in UTMOS (3.66) and Similarity (0.996).

Discussion. These results confirm the importance of carefully balancing modality dropout

Table 6: Ablation of modality dropout probability during latent representation fusion in FuseCodec. **Dropout** indicates the stochastic masking rate applied independently to semantic and contextual representations during training. Moderate dropout prevents over-reliance on a single modality, while higher rates degrade multimodal integration. *A 10% dropout rate achieves the best trade-off, maximizing intelligibility and perceptual quality.*

Model Variant	Dropout	Content Preservation			Speech Naturalness			
		WER↓	WIL↓	STOI↑	ViSQOL↑	PESQ↑	UTMOS↑	Similarity↑
FuseCodec-Fusion	10%	**3.99**	**6.45**	**0.95**	**3.47**	**3.13**	3.63	0.995
FuseCodec-Fusion	30%	4.10	6.63	0.94	3.29	2.96	3.65	0.995
FuseCodec-Fusion	50%	4.09	6.58	0.94	3.33	2.97	**3.66**	**0.996**
FuseCodec-Fusion	70%	4.08	6.64	0.93	3.26	2.91	3.63	0.995
FuseCodec-Fusion	90%	4.15	6.67	0.93	3.26	2.86	3.61	0.995

during latent fusion and underscore the value of semantic-contextual representation integration. Preserving a sufficient portion of the auxiliary representations by using a small 10% dropout rate achieves the most effective use of semantic and contextual information.

As the dropout rate increases, the model receives increasingly less additional modality information, reducing its ability to align latent tokens with multimodal supervision. This negatively affects intelligibility (WER, WIL) and perceptual quality (ViSQOL, PESQ).

Interestingly, metrics such as UTMOS and Similarity remain relatively stable or improve at moderate dropout rates (50%), suggesting that prosodic and speaker characteristics are preserved within the base latent representations. However, the loss of some semantic-contextual information comes at the cost of worse content preservation.

Overall, the findings suggest that light dropout (10%) provides the best trade-off, ensuring robust yet expressive multimodal grounding during latent token fusion.

C.5 Ablation: Quntizer Layer Configuration in Semantic-Contextual Guidance

Setup. We study the impact of RVQ layer supervision depth in the distillation objective. Our method, FuseCodec-Distill, uses **first-layer supervision**, projecting the first-layer RVQ tokens $\mathbf{Q}^{(1)}$ and computing cosine similarity (see Sections 2.3.2 and 2.3.3).

We compare this against an ablated variant, **all-layer supervision**, which averages the outputs from all eight RVQ layers. We define the averaged RVQ output as:

$$\begin{aligned}\mathbf{Q}^{(1:8)} &= \frac{1}{8}\sum_{i=1}^{8}\mathbf{Q}^{(i)} \in \mathbb{R}^{T'\times D},\\ \mathbf{Q}'^{(1:8)} &= \mathbf{Q}^{(1:8)}\mathbf{W}\end{aligned} \tag{25}$$

In the **Global Semantic-Contextual Supervision** setting, we apply the **all-layer supervision** to the distillation loss as:

$$\mathcal{L}_{\text{distill}} = -\frac{1}{T'}\sum_{t=1}^{T'}\log\sigma\left(\frac{1}{2}\left[\cos\left(\mathbf{Q}_t'^{(1:8)},\tilde{\mathbf{S}}_t\right) + \cos\left(\mathbf{Q}_t'^{(1:8)},\tilde{\mathbf{C}}_t\right)\right]\right) \tag{26}$$

Similarly, for the **Temporally Aligned Contextual Supervision** setting, we apply the **all-layer supervision** to the distillation loss as:

$$\mathcal{L}_{\text{align}} = -\frac{1}{T'}\sum_{t=1}^{T'}\log\sigma\left(\cos\left(\mathbf{Q}_t'^{(1:8)},\mathbf{C}_t^{*}\right)\right) \tag{27}$$

Results. Table 7 shows the effect of RVQ supervision depth across both distillation configurations. For FuseCodec (Distill), which uses Global Semantic-Contextual Supervision, first-layer supervision achieves the strongest performance across all content preservation and naturalness metrics, with the lowest WER (4.09), WIL (6.60), and highest STOI (0.94), ViSQOL (3.43), PESQ (3.06), UTMOS (3.65), and Similarity (0.996). Similarly, FuseCodec (ContexAlign), which uses Temporally Aligned Contextual Supervision, First-layer supervision again achieves stronger results in WER (4.15), WIL (6.70), ViSQOL (3.18), PESQ (2.85), and Similarity (0.995). In contrast, using all-layer supervision leads to consistent degradation across most metrics in both settings.

Table 7: Ablation of RVQ supervision depth under global (Distill) and temporally aligned (ContexAlign) guidance. **First Layer** indicates supervision is applied only to the first-layer RVQ tokens, while **All Layers** averages representations from all eight RVQ layers before supervision. *Supervising the first-layer RVQ tokens leads to stronger semantic-contextual grounding and improved intelligibility compared to all-layer supervision.*

Model Variant	RVQ Layer	Content Preservation			Speech Naturalness			
		WER↓	WIL↓	STOI↑	ViSQOL↑	PESQ↑	UTMOS↑	Similarity↑
FuseCodec-ContextAlign	First Layer	**4.15**	**6.70**	0.93	**3.18**	**2.85**	**3.65**	**0.995**
FuseCodec-ContextAlign	All Layers	4.34	7.04	0.93	3.17	2.72	3.65	0.993
FuseCodec-Distill	First Layer	**4.09**	**6.60**	**0.94**	**3.43**	**3.06**	**3.65**	**0.996**
FuseCodec-Distill	All Layers	4.23	6.86	0.93	3.26	2.84	3.61	0.994

Discussion. The results highlight that the layer at which RVQ tokens are supervised significantly impacts the quality of semantic and contextual guidance during distillation. Supervising the first RVQ layer yields stronger performance, as these tokens encode high-level, abstract representations more aligned with semantic intent and global context. This leads to better linguistic grounding and intelligibility, reflected in improved WER, STOI, and ViSQOL scores.

In contrast, deeper RVQ layers capture lower-level acoustic and residual details, which are less suitable for semantic or contextual alignment. Averaging supervision across all layers matches these fine-grained signals with global ones, impacting the alignment objective. This results in performance drop across content preservation and speech naturalness metrics.

Some naturalness metrics, such as UTMOS and Similarity, remain relatively stable with all-layer supervision, suggesting that speaker identity and prosodic features are distributed throughout the RVQ layers. However, these are insufficient for guiding semantic alignment during distillation.

Overall, applying supervision at the first RVQ layer provides a clearer, more semantically grounded signal, leading to better alignment and overall performance in speech reconstruction.

D Related Work

Recent progress in speech and audio generation has been largely driven by advances in discrete representation learning, neural audio codecs, and language model-based synthesis. VQ-VAE (van den Oord et al., 2018) introduced vector quantization in latent spaces to support symbolic modeling of audio, while HuBERT (Hsu et al., 2021) applied masked prediction over cluster-derived labels to learn contextualized speech features in a self-supervised manner. SoundStream (Zeghidour et al., 2022) proposed a causal adversarially trained codec with residual vector quantization (RVQ) and demonstrated scalable compression at low bitrates. HiFi-Codec (Yang et al., 2023) further improved efficiency by introducing group residual quantization, reducing the number of required codebooks while preserving audio fidelity. On the generative side, AudioLM (Borsos et al., 2023) modeled long-range dependencies in semantic and acoustic tokens using transformer-based language modeling. This approach was extended by VALL-E (Wang et al., 2023), which enabled zero-shot text-to-speech synthesis by conditioning on short acoustic prompts and leveraging codec token generation. To improve the suitability of tokenization for language modeling tasks, X-Codec (Ye et al., 2024) integrated semantic embeddings from pretrained models into the quantization pipeline, while LAST (Turetzky and Adi, 2024) learned a tokenizer supervised by a frozen language model to improve downstream ASR and speech generation performance. HiFi-GAN (Kong et al., 2020) introduced multi-period and multi-scale discriminators, enabling high-fidelity waveform synthesis with real-time efficiency.

In parallel, codec designs have evolved to improve training stability and perceptual quality. EnCodec (Défossez et al., 2022) introduced a GAN-based codec architecture with multi-loss balancing and spectrogram-based discrimination, setting a new benchmark for real-time low-bitrate synthesis. BigCodec (Xin et al., 2024) scaled the VQ-VAE framework and showed that a single large codebook could achieve near-human perceptual quality at 1 kbps. DAC (Kumar et al., 2023) proposed refinements to residual quantization, such as factorized and normalized codebooks, and introduced advanced discriminators to improve qual-

ity under bitrate constraints. More recent work has focused on improving token expressiveness for downstream tasks. SpeechTokenizer (Zhang et al., 2024) demonstrated that hierarchical quantization improves resynthesis and zero-shot TTS, while DM-Codec (Ahasan et al., 2024) aligned quantization layers with pretrained speech and text models to reduce WER and enhance contextual fidelity. Finally, NaturalSpeech 3 (Ju et al., 2024) introduced a factorized codec for disentangling prosodic and acoustic attributes in speech, and Moshi (Défossez et al., 2024) unified ASR and TTS in a streaming, full-duplex transformer model operating on jointly learned semantic and acoustic tokens.

E Qualitative Comparison

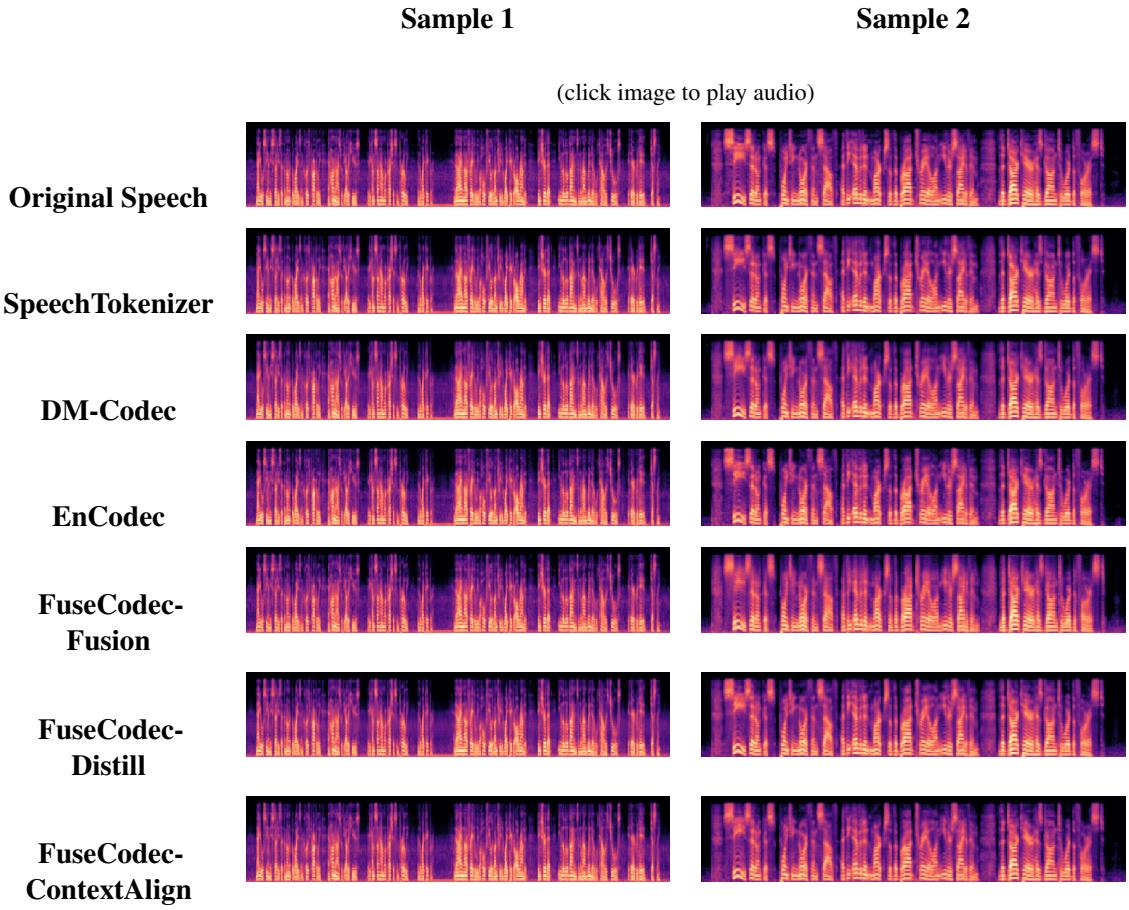

Figure 2: Qualitative speech reconstruction results comparing our method to multiple baselines. Each cell shows the spectrogram output for two samples; clicking an image plays the corresponding audio.