SUBALIGN: Speech Tokenization Aligned with LLM Vocabularies for Spoken Language Modeling

Anonymous submission

Abstract

One factor contributing to the performance discrepancy between large language models and spoken language models is the modality gap in their representations. To address this issue, we introduce SUBALIGN, the first speech tokenization framework to explicitly segment speech at the subword level corresponding to large language model vocabularies. Each resulting SUBALIGN unit is composed of the textual content, acoustic features, and duration associated with its respective subword. Building on this framework, we present SUBALIGN-SLM, a spoken language model trained on SUBALIGN units, and demonstrate the effectiveness of SUBALIGN on downstream tasks. Extensive automatic and human evaluations show that SUBALIGN-SLM surpass baseline models, demonstrating the potential of SUBALIGN for speech processing applications.

1 Introduction

Large language models (LLMs) have demonstrated remarkable progress in language processing capabilities, especially in generative tasks. Recent studies have sought to extend the capabilities of LLMs to spoken language models (SLMs), which jointly model the acoustic and semantic aspects of spoken language (Lakhotia et al. 2021). However, SLMs often fall behind LLMs in producing semantically coherent outputs (Wang et al. 2024; Choi et al. 2024). This performance discrepancy stems from the modality gap between speech and text, including the differences in sequence length distribution and the presence of rich paralinguistic information.

Inspired by the use of discrete representations in LLMs, Lakhotia et al. (2021) propose training SLMs using discrete speech tokens computed from raw audio. However, since acoustic signal is typically segmented at a finer temporal resolution than text, speech tokenization often result in greater number of tokens per utterance compared to text. This increases the challenge of modeling long-range dependencies, leading to performance degradation in SLMs.

Recent studies sought to address this problem by reducing the length of the speech representation sequence and observed some performance improvements (Cho et al. 2025; Baade, Peng, and Harwath 2025).Although these approaches successfully lower the temporal resolution of speech frames, they do not account for the vocabularies of the downstream LLMs, which may limit their effectiveness in downstream applications. Tseng et al. (2025), on the other hand, introduce a text-aligned speech tokenization approach that fixes the number of speech tokens to the corresponding LLM token length. However, their method relies on word-level alignments and duplicates word-level representations to subwords, which does not fully exploit subword-level granularity and may result in the loss of fine-grained acoustic or paralinguistic information.

In this work, we propose SUBALIGN, the first speech tokenization framework that determines speech token boundaries based on LLM subword vocabularies. Our approach first aligns speech frames to individual characters using monotonic alignment search, then aggregates the corresponding frames to construct subword-level representations. This aggregation method enables the framework to accommodate infrequent words or tokens effectively.

Experimental results on speech reconstruction show that the SUBALIGN framework outperforms speech tokenizers with similar bitrates. Furthermore, SUBALIGN-SLM, an SLM trained on SUBALIGN units, achieves strong performance on likelihood-based classification tasks and surpasses baseline models in speech continuation, exhibiting superior acoustic consistency and semantic coherence.
Our key contributions are as follows:

- We present SUBALIGN, the first speech tokenization framework to explicitly segment speech at the subword level corresponding to LLM vocabularies.
- SUBALIGN-SLM exhibits superior performance compared to the baseline models, demonstrating the promise of SUBALIGN for speech processing applications.
- We improve the stability of speech-subword alignment with a two-step approach: first, computing character-level speech alignments, then aggregating them to the subword level.

2 Related Work

Spoken Language Models SLMs have followed the trends established by LLMs, adopting next token prediction as the training objective. Early SLMs (Lakhotia et al. 2021; Kharitonov et al. 2022; Borsos et al. 2023) bypass textual supervision entirely, directly generating audio from acoustic cues. While these approaches capture prosody and speaker

characteristics, they often struggle with maintaining long-range semantic coherence. To address these limitations, recent work leverages textual pretraining (Hassid et al. 2023; Défossez et al. 2024) or joint speech-text modeling (Nguyen et al. 2025; Gao et al. 2025), substantially improving semantic consistency and fluency. Despite these advances, SLMs still lag behind LLMs in generating coherent, contextually appropriate output.

Speech Tokenization Speech tokenization is a critical step for spoken language modeling. Recent methods discretize speech into tokens, often using vector quantization. Early approaches adopted frame-level tokens (Hsu et al. 2021; Lakhotia et al. 2021; Hassid et al. 2023), which preserve phonetic detail but produce long sequences that hinder efficient modeling. To reduce sequence length, some approaches (Cho et al. 2025; Baade, Peng, and Harwath 2025) compress speech into syllable-level units, though these remain misaligned with LLM vocabularies and still lack semantic information. Text-aligned methods (Tseng et al. 2025) address this by mapping speech to ASR vocabularies. However, due to mismatches between ASR and LLM vocabularies, this approach compresses acoustic tokens at the word level, potentially sacrificing granularity in the acoustic representation.

Speech-Text Alignment Speech-text alignment has long been a critical challenge in both automatic speech recognition (ASR) and text-to-speech (TTS) research. Early approaches addressed this problem using statistical models such as Gaussian Mixture Models and Hidden Markov Models to model the correspondence between speech and text (McAuliffe et al. 2017; Duan et al. 2013). More recently, ASR models use Connectionist Temporal Classification (CTC) loss (Graves et al. 2006). Using CTC loss, ASR models align audio frames with text by marginalizing over all possible alignments without requiring pre-segmented data. In TTS systems (Kim et al. 2020; Kim, Kong, and Son 2021), alignment estimation is often achieved via *monotonic alignment search*, a dynamic programming technique that finds the most probable monotonic path between text and speech features. Similar to this, Badlani et al. (2022) proposed a unified framework for alignment learning based on forward-sum and Viterbi algorithms with static priors, resulting in improved alignment robustness and synthesis quality.

3 Approach: SUBALIGN

Our framework, SUBALIGN, tokenizes speech so that the temporal boundaries of the speech units align with the subword boundaries in the text, ensuring that each speech token corresponds to a single subword token. SUBALIGN consists of two components: (a) *Speech Tokenization*, which converts input speech into a sequence of subword-aligned discrete units; and (b) *Speech Reconstruction*, which restores the original speech from these units. Figure 1 shows the overview of the framework.

3.1 Speech Tokenization

Aligning speech directly with the subword vocabularies of modern LLMs presents significant challenges, due to the extensive vocabulary size and the need for a large speech dataset that covers all vocabulary items. To address these issues, our framework proposes an alternative approach that aligns speech with individual characters and aggregates character-level representations to derive subword-level representations. This approach assumes a monotonic alignment between speech and subwords or characters.

The tokenization process consists of four steps: (i) *Speech Encoding* for extracting frame-level features, (ii) *Speech-Character Alignment* for mapping frames to characters, (iii) *Subword-Level Aggregation* for pooling character-level features into subword representations, and (iv) *Quantization* for discretizing subword embeddings via residual vector quantization (RVQ).

Speech Encoding We use a pretrained speech encoder to extract frame-level representations from the raw audio input $\boldsymbol{x}$. The encoder produces two types of features for each frame: semantic features $\boldsymbol{x}^s = (x_1^s, \ldots, x_T^s)$ and acoustic features $\boldsymbol{x}^a = (x_1^a, \ldots, x_T^a)$. The semantic features $\boldsymbol{x}^s$ correspond to the final hidden states at each timestep of the speech encoder outputs. The acoustic features $\boldsymbol{x}^a$ are computed by concatenating the hidden states from the lower layers of the speech encoder:

$$\boldsymbol{x}^s, \boldsymbol{x}^a = \textsc{SpeechEncoder}(\boldsymbol{x}; \theta). \tag{1}$$

By separating the two features, our framework can utilize high-level semantic information for alignment computation while preserving the acoustic details essential for tasks such as speech reconstruction.

Speech-Character Alignment To obtain character-level boundaries of speech frames, we first generate the corresponding transcript $\boldsymbol{\tau}$ using an ASR model, resulting in a sequence of characters $\boldsymbol{\tau} = (c_1, \ldots, c_L)$. For each semantic feature x_t^s, we use a linear classifier to predict the probability logits over the character set $\mathcal{C}$. The classifier is trained with the CTC objective, which maximizes the likelihood of the alignment between the speech and the character sequence.

Given the target character sequence $\boldsymbol{c} = (c_1, \ldots, c_L)$, the set of all valid monotonic alignments is defined as:

$$\mathcal{A}(\boldsymbol{c}, T) = \left\{ \pi \in \{1, \ldots, L\}^T \mid \pi_{t+1} \geq \pi_t \right\}. \tag{2}$$

The CTC loss is computed as:

$$\mathcal{L}_{\text{CTC}} = -\log \sum_{\pi \in \mathcal{A}(\boldsymbol{c}, T)} \prod_{t=1}^{T} p_\theta \left(c_{\pi_t} \mid x_t^s \right). \tag{3}$$

During inference, we compute the most probable frame-to-character alignment by applying the Viterbi algorithm:

$$\pi^* = \arg\max_{\pi \in \mathcal{A}(\boldsymbol{c}, T)} \sum_{t=1}^{T} \log p_\theta \left(c_{\pi_t} \mid x_t^s \right). \tag{4}$$

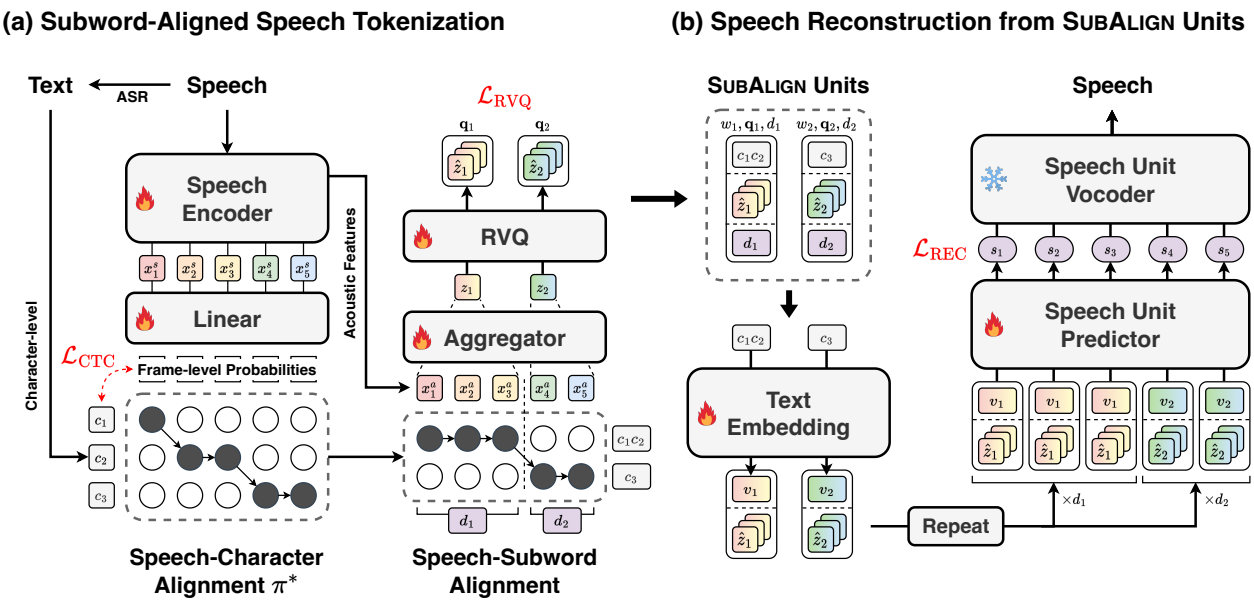

Figure 1: **Overview of SubAlign:** (a) Raw speech is processed through the speech encoder to extract frame-wise semantic and acoustic features ($\boldsymbol{x}^s, \boldsymbol{x}^a$). First, character-level alignment is computed using semantic features via CTC-based monotonic alignment search, and the aligned segments are then matched to LLM subwords. The corresponding acoustic features are aggregated via attention-based pooling and discretized through Residual Vector Quantization (RVQ), producing SubAlign units $(w_n, \mathbf{q}_n, d_n)$ that contain subword text, quantized acoustic codes, and duration. (b) Speech is reconstructed from SubAlign units with two-stage decoding. First, subword embeddings (v_n) are computed and concatenated with quantized acoustic embeddings $(\hat{z}_n)$, then repeated according to their durations (d_n) to form frame-level representations. These are subsequently converted into target speech units and synthesized into waveforms using a pretrained speech unit vocoder.

Subword-Level Aggregation After obtaining the frame-to-character alignment π^*, we determine subword boundaries by grouping consecutive characters according to the subword tokenizer. For each subword $w_n = (w_n^{(1)}, \ldots, w_n^{(\ell_n)})$, where $w_n^{(i)}$ is the i-th character and ℓ_n is the length of subword w_n, we define the corresponding set of frame indices as:

$$\mathcal{T}_n = \left\{ t \mid \sum_{i=1}^{n-1} \ell_i \leq \pi_t^* < \sum_{i=1}^{n} \ell_i \right\}. \tag{5}$$

We apply attention-based pooling on the acoustic features $(x_t^a)_{t \in \mathcal{T}_n}$ to get the subword-level representations. Specifically, we prepend a learnable `[AGG]`$^{\mathrm{T}}$ token to the features and process the sequence using a shallow Transformer encoder. The final hidden state of the aggregate token serves as the subword-level embedding:

$$z_n = \textsc{Transformer}\left(\texttt{[AGG]}^{\mathrm{T}}, (x_t^a)_{t \in \mathcal{T}_n}; \theta\right). \tag{6}$$

This process yields a sequence of subword embeddings $\mathbf{z} = (z_1, \ldots, z_N)$, where each $z_n \in \mathbb{R}^D$ represents the acoustic content of subword w_n.

Residual Vector Quantization Each subword acoustic features z_n are discretized using RVQ. The RVQ module consists of R stages. At each stage r, a codebook with vocabulary size K is used to quantize the residual of the previous stage. This produces code indices $\mathbf{q}_n = (q_n^{(1)}, \ldots, q_n^{(R)})$, where $q_n^{(r)} \in \{1, \ldots, K\}$. Subsequently, we can approximate the original embedding by summing all vectors corresponding to the code indices at each stage:

$$\hat{z}_n^{(r)} = \textsc{Codebook}_r(q_n^{(r)}), \quad \hat{z}_n = \sum_{r=1}^{R} \hat{z}_n^{(r)} \tag{7}$$

Both the discrete code sequences $\mathbf{q} = (\mathbf{q}_1, \ldots, \mathbf{q}_N)$ and the quantized embeddings $\hat{\mathbf{z}} = (\hat{z}_1, \ldots, \hat{z}_N)$ are temporally aligned with the subword sequence. The codebook weights are updated using exponential moving average (EMA), and the quantizer is trained by minimizing the commitment loss:

$$\mathcal{L}_{\text{RVQ}}(\theta) = \frac{1}{N} \sum_{n=1}^{N} \|z_n - \text{sg}[\hat{z}_n]\|_2^2 \tag{8}$$

where $\text{sg}[\cdot]$ is the stop-gradient operator to prevent gradients from flowing into the codebooks.

After tokenization, each SubAlign unit is represented as a tuple $(w_n, \mathbf{q}_n, d_n)$. Here, w_n denotes a subword, $\mathbf{q}_n = (q_n^{(1)}, \ldots, q_n^{(R)})$ is the R-tuple of quantized acoustic code indices, and $d_n = |\mathcal{T}_n|$ represents the number of acoustic frames associated with w_n.

3.2 Speech Reconstruction

Instead of directly generating waveforms from SubAlign units, we adopt a two-stage strategy: we first predict inter-

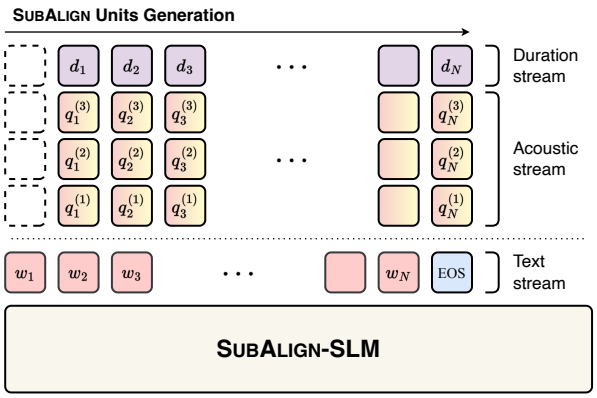

Figure 2: **Overview of SUBALIGN-SLM.** The model autoregressively generates SUBALIGN units with a delayed prediction pattern. It is initialized from a pretrained LLM and extended with modality-specific heads for predicting subwords, acoustic tokens, and durations.

mediate frame-level speech unit representations and subsequently decode them into waveforms using a pretrained speech unit vocoder.

To enable frame-wise prediction, we construct a frame-level input sequence from subword-level SUBALIGN units. For each subword $w_n = (w_n^{(1)}, \ldots, w_n^{(\ell_n)})$, we first obtain its semantic representation by aggregating the embeddings of its individual characters using a Transformer encoder with an attention-based pooling mechanism:

$$v_n = \textsc{Transformer}\left(\texttt{[AGG]}^{\mathrm{R}}, \texttt{Embed}(w_n^{(i)})_{i=1}^{\ell_n}; \theta\right), \tag{9}$$

where $\texttt{[AGG]}^{\mathrm{R}}$ is a learnable aggregation token prepended to the character sequence.

Next, we concatenate the subword embedding v_n with its corresponding quantized acoustic representation $\hat{z}_n$, and repeat the resulting vector d_n times to match the desired frame-level resolution:

$$\boldsymbol{u} = \left([v_1 \oplus \hat{z}_1]^{\times d_1}, \ldots, [v_N \oplus \hat{z}_N]^{\times d_N}\right). \tag{10}$$

The resulting sequence $\boldsymbol{u}$ is fed into a Transformer encoder-based Speech Unit Predictor, which predicts the target frame-level speech units $\boldsymbol{s} = (s_1, \ldots, s_T)$:

$$\hat{\boldsymbol{s}} = \textsc{SpeechUnitPredictor}(\boldsymbol{u}; \theta). \tag{11}$$

Finally, the predicted sequence $\hat{\boldsymbol{s}}$ is passed through a pretrained vocoder to reconstruct the waveform. The model is trained using a cross-entropy loss between the predicted distribution and the ground-truth speech unit tokens:

$$\mathcal{L}_{\text{REC}}(\theta) = -\frac{1}{T}\sum_{t=1}^{T} \log p_\theta(s_t \mid \boldsymbol{u}). \tag{12}$$

3.3 Training Objective

All modules are trained jointly in an end-to-end manner. The training objective is the sum of three losses:

$$\mathcal{L}(\theta) = \mathcal{L}_{\text{CTC}}(\theta) + \mathcal{L}_{\text{RVQ}}(\theta) + \mathcal{L}_{\text{REC}}(\theta). \tag{13}$$

3.4 SUBALIGN-SLM

Building on the SUBALIGN framework, we introduce SUBALIGN-SLM, a spoken language model which operates on sequences of SUBALIGN units. SUBALIGN-SLM is an autoregressive language model finetuned to follow a delayed prediction pattern; the model generates acoustic codes and duration for one step after producing the corresponding subword token. Specifically, for the n-th unit in the sequence, the model predicts the subword w_n at the n-th generation step, and then predicts the acoustic code q_n and the duration d_n at the $(n+1)$-th step. Figure 2 illustrates this process.

We initialize SUBALIGN-SLM from a pretrained LLM and incorporate modality-specific prediction heads: one for subwords, R for acoustic codes, and one for duration. The model is trained using the sum of cross-entropy losses over all prediction targets:

$$\begin{aligned}\mathcal{L}_{\text{SLM}} = &-\frac{1}{N}\sum_{n=1}^{N} \log p_{\text{SLM}}(w_n \mid w_{<n}, \mathbf{q}_{<n-1}, d_{<n-1}) \\ &-\frac{1}{N}\sum_{n=1}^{N} \log p_{\text{SLM}}(q_n^{(*)}, d_n \mid w_{\leq n}, \mathbf{q}_{<n}, d_{<n}).\end{aligned} \tag{14}$$

4 Experiment Setup

4.1 Implementation

We use `Whisper-large-v3` (Radford et al. 2023) to obtain transcripts from speech data. We adopt the pretrained `Data2vec2.0-base` (Baevski et al. 2022) as the speech encoder, which consists of a CNN feature extractor followed by an 8-layer Transformer encoder with 12 attention heads and 768 hidden dimensions. We use the hidden states from the first four layers as the acoustic features. The subword-level aggregator is implemented as a 2-layer Transformer encoder. For discretizing the acoustic features, we use an RVQ with three stages ($R = 3$) and a codebook size of 512 at each stage.

For speech reconstruction, we adopt S^3 tokens from CosyVoice (Du et al. 2024) as the target speech units, and extract speaker embeddings using the same approach. The speech unit predictor is a 6-layer Transformer encoder with 8 attention heads and 768 hidden dimensions. The same architecture is used for character aggregation in the subword-level aggregator.

For SUBALIGN-SLM, we experiment with two backbone LLMs: Llama-3.2-1B (Grattafiori et al. 2024) and Qwen3-1.7B (Yang et al. 2025). The SUBALIGN-SLM (*punc*) refers to the setting where the ASR model is prompted to include punctuations in the transcripts.

Training We train both the tokenizer and the SLMs using two datasets: the English subset of Emilia (He et al. 2024) (40,000 hours) and the LibriTTS train set (Zen et al. 2019) (565 hours). Emilia provides web-scale speech data with pseudo-transcripts, while LibriTTS offers clean, read-style recordings.

Model	Bitrate ↓	Quality		Similarity			
		WER ↓	UTMOS ↑	SECS ↑	F0 RMSE ↓	STOI ↑	DC ↑
Mel + BigVGAN	–	4.23	3.73	0.867	57.13	0.992	0.996
SpeechTokenizer	4000	5.06	3.57	0.820	67.25	0.909	0.982
SpeechTokenizer	2000	7.04	3.28	0.639	71.21	0.866	0.977
SpeechTokenizer	1000	10.90	2.21	0.325	89.49	0.765	0.969
Mimi	1000	7.28	3.37	0.633	66.85	0.897	0.978
WavTokenizer	900	7.82	3.69	0.711	68.08	0.905	0.981
S^3 Tokens	600	6.48	3.77	0.649	71.09	0.779	0.972
Sylber *(w/o quant.)*	–	8.76	3.99	0.369	72.49	0.737	0.958
TASTE	195	9.69	3.94	0.585	79.00	0.431	0.928
Ours *(w/o quant.)*	–	7.01	4.01	0.610	71.90	0.740	0.969
Ours	193	5.66	4.05	0.587	73.89	0.686	0.959

Table 1: **Comparison of reconstruction and perceptual metrics.** We report Bitrate, WER, UTMOS, SECS, F0 RMSE, STOI, and DC. Lower is better for ↓ metrics; higher is better for ↑ metrics. '*w/o quant.*' indicates models using continuous tokens.

We first train the SUBALIGN tokenization module for 500K steps using the target LLM vocabulary. Following Cho et al. (2025), we apply noise augmentation to the training data, using the dataset introduced by Reddy et al. (2021), to improve the robustness of the tokenization. All SLMs are trained with Low-Rank Adaptation (LoRA) (Hu et al. 2022) applied to all linear layers, using rank $r = 64$ and scaling factor $\alpha = 128$. We train our SLMs for 200K steps.

4.2 Setup for Speech Reconstruction

We evaluate the reconstruction performance of the SUBALIGN on the LibriTTS *test-clean* and *test-other* splits, aiming to preserve both semantic and acoustic information from the original speech. To evaluate the intelligibility and naturalness of the reconstructed speech, we compute the word error rate (WER) using `NeMo` (Kuchaiev et al. 2019) and UTMOS (Saeki et al. 2022). Pitch accuracy is evaluated using the root-mean-square error of fundamental frequency (F0 RMSE). Speaker consistency is measured with speaker embedding cosine similarity (SECS), computed using ESPNet-SPK (Jung et al. 2024). In addition, we report short-time objective intelligibility (STOI) to assess the intelligibility of the reconstructed speech. To check duration consistency (DC), we first find the word-level segments of the original and reconstructed speech using the Montreal Forced Aligner (McAuliffe et al. 2017), and compute what fraction of words have matching durations between them, allowing for a preset tolerance window of 100ms.

Baselines We compare SUBALIGN with several baselines, including the BigVGAN-reconstructed mel-spectrogram as a topline (Lee et al. 2023), as well as various speech tokenization models such as SpeechTokenizer (Zhang et al. 2024) with multiple bitrate configurations, Mimi (Défossez et al. 2024), WavTokenizer (Ji et al. 2025), Sylber (Cho et al. 2025), TASTE (Tseng et al. 2025), and S^3 tokens (Du et al. 2024).

4.3 Setup for Spoken Language Modeling

We evaluate our SLMs on two tasks: likelihood-based classification and speech continuation, to assess both their acoustic and semantic understanding.

Likelihood-Based Classification As per convention in SLM evaluation, we assess the language modeling capabilities of SUBALIGN-SLM through likelihood-based classification tasks (Hassid et al. 2023; Nguyen et al. 2025; Tseng et al. 2025), where a model receives the speech context along with two possible continuations and selects the candidate with the higher likelihood as the answer. For evaluation, we use SALMon (Maimon, Roth, and Adi 2025) and spoken StoryCloze (Hassid et al. 2023) benchmark. From the SALMon benchmark, we evaluate acoustic properties of speech such as sentiment consistency, speaker consistency, and gender consistency. We also perform an auxiliary evaluation set for energy consistency using a similar process as SALMon; we sample 200 examples from the VCTK dataset (Yamagishi et al. 2012), normalize the volume, and create negative examples by reducing the amplitude of either the first or second half of the audio to 10–30% of its original level. Spoken StoryCloze, which consists of sStoryCloze and tStoryCloze, evaluates the model's ability to detect semantic inconsistencies in the content (Hassid et al. 2023).

Speech Continuation To evaluate conditional speech generation, we follow the protocol of TASTE (Tseng et al. 2025). We provide the model with a 3-second speech segment from the LibriSpeech test set and prompt it to generate the subsequent continuation of the speech (Panayotov et al. 2015). We evaluate the quality of the generated speech using both automatic and human evaluation. Specifically, we use GPT-4o for assessing the semantic relevance, UTMOS for the perceptual acoustic quality, and SECS for speaker consistency of the generated content. Before submitting it to GPT-4o, we transcribe the generated speech using `Whisper-large-v3` (Radford et al. 2023). Moreover, we conduct human listening tests to evaluate the overall consistency of the generated continuation. Detailed evaluation protocols and instructions are provided in Appendix C.

Method	Backbone	Size	Acoustic					Semantic		
			Sentiment	Speaker	Gender	Energy	Avg.	sSC	tSC	Avg.
ASR + LLM	Llama-3.2	1B	–	–	–	–	–	66.2	<u>90.3</u>	<u>78.2</u>
ASR + LLM	Qwen3	1.7B	–	–	–	–	–	65.6	85.7	75.7
TWIST	OPT	1.3B	61.5	69.0	69.5	60.0	65.0	52.4	70.6	61.5
TWIST	Llama	7B	61.5	71.0	70.0	<u>61.5</u>	66.0	55.3	74.1	64.7
SpiRit-LM	Llama-2	7B	54.5	69.5	67.0	<u>61.5</u>	63.1	61.0	82.9	72.0
SpiRit-LM (*expr.*)	Llama-2	7B	**73.5**	**81.0**	**85.0**	49.0	**72.1**	56.9	75.4	66.2
TASLM (*embed.*)	Llama-3.2	1B	57.5	67.0	75.5	–	–	64.0	89.5	76.7
TASLM (*token*)	Llama-3.2	1B	59.0	68.0	70.5	50.0	61.9	64.2	88.9	76.5
Ours										
SUBALIGN-SLM	Llama-3.2	1B	<u>67.0</u>	69.5	77.0	**62.0**	68.9	63.9	89.0	76.5
SUBALIGN-SLM (*punc*)	Llama-3.2	1B	65.5	<u>75.0</u>	77.5	61.0	<u>69.8</u>	**67.7**	**91.5**	**79.6**
SUBALIGN-SLM	Qwen3	1.7B	65.5	66.5	<u>78.5</u>	58.0	67.1	<u>66.5</u>	87.7	77.1

Table 2: **Results of different SLMs on SALMon and StoryCloze.** We report likelihood-based accuracy on SALMon (acoustic aspect) and StoryCloze (semantic aspect). The best scores are highlighted in **bold**, and the second-best scores are <u>underlined</u>.

Method	Backbone	Size	GPT-4o	UTMOS	SECS	Human Eval
ASR + LLM + TTS	Llama-3.2	1B	2.54 ± 0.20	$\mathbf{3.60 \pm 0.14}$	0.609 ± 0.021	4.020 ± 0.120
ASR + LLM + TTS	Qwen3	1.7B	2.40 ± 0.20	3.58 ± 0.13	0.596 ± 0.022	3.717 ± 0.140
TWIST	OPT	1.3B	1.96 ± 0.12	3.58 ± 0.12	–	–
TWIST	Llama	7B	2.23 ± 0.16	3.38 ± 0.16	–	–
SpiRit-LM	Llama-2	7B	2.45 ± 0.22	3.30 ± 0.05	–	–
SpiRit-LM (*expr.*)	Llama-2	7B	1.87 ± 0.14	3.20 ± 0.08	–	–
TASLM (*token*)	Llama-3.2	1B	2.73 ± 0.18	3.54 ± 0.11	0.556 ± 0.024	3.220 ± 0.160
Ours						
SUBALIGN-SLM	Llama-3.2	1B	$\mathbf{3.07 \pm 0.19}$	3.38 ± 0.14	$\mathbf{0.642 \pm 0.025}$	$\mathbf{4.187 \pm 0.105}$
SUBALIGN-SLM	Qwen3	1.7B	2.97 ± 0.21	3.31 ± 0.14	0.636 ± 0.024	4.003 ± 0.120

Table 3: **Speech continuation results across different SLMs.** We report scores for semantic quality (GPT-4o), acoustic quality (UTMOS), and consistency with the prompt waveform (SECS and Human Evaluation) of the continuation. Higher values indicate better performance. For SECS and Human Evaluation, results are only available for methods with access to prompt-consistent generation.

Baselines We compare our model with several SLMs, including TWIST (Hassid et al. 2023), SpiRit-LM (Nguyen et al. 2025), and TASLM (Tseng et al. 2025). We also include cascaded systems, which employ an ASR model, `Whisper-large-v3`, followed by an LLM and a TTS model, CosyVoice, in sequence.

5 Results and Discussion

5.1 Results on Speech Reconstruction

Table 1 compares the reconstruction quality of different tokenization methods. SUBALIGN achieves the lowest bitrate among all approaches, compressing speech to 193 bps. It also achieves superior perceptual quality in the reconstructed output, showing the highest UTMOS score among the baselines. In terms of WER, SUBALIGN performs comparably to SpeechTokenizer, despite operating at only one-twentieth of the bitrate. This indicates the effectiveness of our tokenization in producing speech that is both perceptually natural and semantically intelligible.

Results from similarity metrics demonstrate that our model effectively preserves prosodic cues, speaker identity, and temporal structure. While higher-bitrate tokenizers capture finer acoustic details, our approach outperforms Sylber and TASTE, models at similar bitrates. Notably, compared to SpeechTokenizer at a bitrate of 1000, our model achieves comparable or superior preservation of acoustic details at a much lower bitrate. This highlights our method's ability to encode rich acoustic information using fewer bits efficiently.

5.2 Results on Spoken Language Modeling

In the likelihood-based classification benchmarks, shown in Table 2, our model demonstrates robust performance across the board. The baseline models tend to favor either acoustic consistency or semantic consistency, but not both. For instance, SpiRit-LM (*expr.*) and TASLM (*token*) are the best-performing models in their respective categories, yet each performs poorly in the alternate category. On the contrary, the SUBALIGN-SLM models achieve top performance in both acoustic and semantic consistency. In fact, SUBALIGN

Ablation	Bitrate ↓	WER ↓	UTMOS ↑	SECS ↑
SUBALIGN (*default*)	193	5.66	4.05	0.587
w/ Qwen vocab.	193	5.83	4.05	0.583
w/o char-alignment	193	24.57	3.63	0.471
w/ subword emb.	193	23.24	3.97	0.471
$R = 1$	95	5.75	4.05	0.458
$R = 4$	226	5.87	4.05	0.593
$R = 8$	358	5.82	4.05	0.605

Table 4: **Ablation study on SubAlign modules and quantization levels.** We report Bitrate, WER, UTMOS, and SECS.

(*punc*) variant surpasses even the topline ASR + LLM baselines in semantic consistency. These findings indicate that our methodology effectively balances acoustic and semantic representational capabilities.

As shown in Table 3, SUBALIGN-SLM also demonstrates strong performance in speech continuation evaluation. Notably, with both LLM backbones, SUBALIGN-SLM achieves the highest GPT-4o scores, indicating its ability to generate speech continuations that are semantically consistent to the provided context. It also yields the best scores in speaker consistency and human evaluations, suggesting that the generated speech is acoustically faithful to the prompt as well. Although the UTMOS scores are slightly lower than those of the best-performing models, the difference is not statistically significant, as indicated by the overlapping confidence intervals. These results collectively highlight the robustness and reliability of SUBALIGN-SLM in generating consistent speech outputs.

5.3 Ablation Study

We conduct a series of ablation studies to investigate the key factors influencing the performance of SUBALIGN. Table 4 presents the results of these experiments.

LLM Vocabulary We assess the generalizability of our approach to various LLM vocabulary sets by adapting SUBALIGN to Qwen3 vocabularies. The results indicate no significant difference in performance between the original and Qwen3 vocabulary variant, demonstrating that our method maintains its effectiveness regardless of the underlying vocabulary sets.

Character-Level Alignment To evaluate the intermediate character-speech alignment step, we conduct an ablation study that aligns speech directly to subwords, omitting the character-level alignment. The significant drop in reconstruction quality indicates that the framework struggles in accurately aligning speech to subwords without the intermediate character-level step.

Character-Level Embedding Aggregation We also examine the effectiveness of computing subword embeddings by aggregating character-level embeddings. Specifically, we replace the character embedding aggregation described in Eq.(9) with an embedding layer output $v_n = \texttt{Embed}(w_n)$. This results in drastic increase in WER, suggesting that the character-level embedding aggregation is essential for accurate pronunciation modeling.

Quantization Level We also investigate the effect of the number of RVQ stages on reconstruction quality. Although increasing the number of stages improves speaker consistency, the gap is marginal. Moreover, no significant changes are observed in other metrics. Therefore, as trade-off exists between bitrates and acoustic details, we find our choice of $R = 3$ is a reasonable balance between achieving low bitrates and preserving sufficient acoustic details.

5.4 Limitations

While our method shows promising results, there are several limitations that warrant further investigation. First, the tokenization framework relies on the availability of transcripts, which may constrain applicability in low-resource speech scenarios. Second, all experiments are conducted exclusively on English speech data. It remains to be verified whether the proposed approach generalizes to multilingual or code-switched settings. Lastly, while the evaluation focuses on read-style and web-scale data, the performance of our model in conversational or spontaneous speech remains unexplored. Addressing these limitations will be intriguing directions for future research.

6 Conclusion

We introduce SUBALIGN, a speech tokenization framework that segments speech into subword-level units aligned with LLM vocabularies for spoken language modeling. Unlike previous approaches that represent speech at the frame level or aggregate it at the word level, SUBALIGN computes monotonic character level alignment using text transcripts to precisely group speech frames into subword segments. This alignment enables efficient joint modeling of speech and text while preserving both semantic content and acoustic detail. Experimental results show that SUBALIGN supports high-quality speech reconstruction even at low bitrates and significantly improves spoken language modeling performance in likelihood-based and speech continuation benchmarks.

References

Baade, A.; Peng, P.; and Harwath, D. 2025. SyllableLM: Learning Coarse Semantic Units for Speech Language Models. In *The Thirteenth International Conference on Learning Representations*.

Badlani, R.; Łańcucki, A.; Shih, K. J.; Valle, R.; Ping, W.; and Catanzaro, B. 2022. One TTS alignment to rule them all. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6092–6096. IEEE.

Baevski, A.; Hsu, W.-N.; Xu, Q.; Babu, A.; Gu, J.; and Auli, M. 2022. Data2vec: A general framework for self-supervised learning in speech, vision and language. In *International conference on machine learning (ICML)*, 1298–1312. PMLR.

Borsos, Z.; Vincent, D.; Balahur, A.; Weiss, R.; and et al. 2023. AudioLM: a Language Modeling Approach to Audio Generation. arXiv preprint arXiv:2209.03143.

Cho, C. J.; Lee, N.; Gupta, A.; Agarwal, D.; Chen, E.; Black, A.; and Anumanchipalli, G. 2025. Sylber: Syllabic Embedding Representation of Speech from Raw Audio. In *The Thirteenth International Conference on Learning Representations*.

Choi, K.; Pasad, A.; Nakamura, T.; Fukayama, S.; Livescu, K.; and Watanabe, S. 2024. Self-Supervised Speech Representations are More Phonetic than Semantic. In *Proceedings of Interspeech*, 4578–4582.

Défossez, A.; Mazaré, L.; Orsini, M.; Royer, A.; Pérez, P.; Jégou, H.; Grave, E.; and Zeghidour, N. 2024. Moshi: a speech-text foundation model for real-time dialogue. *arXiv preprint arXiv:2410.00037*.

Du, Z.; Chen, Q.; Zhang, S.; Hu, K.; Lu, H.; Yang, Y.; Hu, H.; Zheng, S.; Gu, Y.; Ma, Z.; et al. 2024. Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens. *arXiv preprint arXiv:2407.05407*.

Duan, Z.; Fang, H.; Li, B.; Sim, K. C.; and Wang, Y. 2013. The NUS sung and spoken lyrics corpus: A quantitative comparison of singing and speech. In *2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, 1–9.

Gao, H.; Shao, H.; Wang, X.; Qiu, C.; Shen, Y.; Cai, S.; Shi, Y.; Xu, Z.; Long, Z.; Zhang, Y.; et al. 2025. Lucy: Linguistic understanding and control yielding early stage of her. *arXiv preprint arXiv:2501.16327*.

Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Graves, A.; Fernández, S.; Gomez, F.; and Schmidhuber, J. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *International conference on machine learning (ICML)*, 369–376.

Hassid, M.; Remez, T.; Nguyen, T. A.; Gat, I.; Conneau, A.; Kreuk, F.; Copet, J.; Defossez, A.; Synnaeve, G.; Dupoux, E.; et al. 2023. Textually pretrained speech language models. *Advances in Neural Information Processing Systems*, 36: 63483–63501.

He, H.; Shang, Z.; Wang, C.; Li, X.; Gu, Y.; Hua, H.; Liu, L.; Yang, C.; Li, J.; Shi, P.; et al. 2024. Emilia: An extensive, multilingual, and diverse speech dataset for large-scale speech generation. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, 885–890. IEEE.

Hsu, W.-N.; Bolte, B.; Tsai, Y.-H. H.; Lakhotia, K.; Salakhutdinov, R.; and Mohamed, A. 2021. HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units. In *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W.; et al. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2): 3.

Ji, S.; Jiang, Z.; Wang, W.; Chen, Y.; Fang, M.; Zuo, J.; Yang, Q.; Cheng, X.; Wang, Z.; Li, R.; Zhang, Z.; Yang, X.; Huang, R.; Jiang, Y.; Chen, Q.; Zheng, S.; and Zhao, Z. 2025. WavTokenizer: an Efficient Acoustic Discrete Codec Tokenizer for Audio Language Modeling. In *The Thirteenth International Conference on Learning Representations*.

Jung, J.-w.; Zhang, W.; Shi, J.; Aldeneh, Z.; Higuchi, T.; Theobald, B.-J.; Abdelaziz, A. H.; and Watanabe, S. 2024. ESPnet-SPK: Full pipeline speaker embedding toolkit with reproducible recipes, self-supervised front-ends, and off-the-shelf models. *arXiv preprint arXiv:2401.17230*.

Kharitonov, E.; Lee, A.; Polyak, A.; Adi, Y.; Copet, J.; Lakhotia, K.; Nguyen, T. A.; Riviere, M.; Mohamed, A.; Dupoux, E.; et al. 2022. Text-Free Prosody-Aware Generative Spoken Language Modeling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 8666–8681.

Kim, J.; Kim, S.; Kong, J.; and Yoon, S. 2020. Glow-tts: A generative flow for text-to-speech via monotonic alignment search. *Advances in Neural Information Processing Systems*, 33: 8067–8077.

Kim, J.; Kong, J.; and Son, J. 2021. Conditional Variational Autoencoder with Adversarial Learning for End-to-End Text-to-Speech. In *International conference on machine learning (ICML)*, 5530–5540.

Kuchaiev, O.; Li, J.; Nguyen, H.; Hrinchuk, O.; Leary, R.; Ginsburg, B.; Kriman, S.; Beliaev, S.; Lavrukhin, V.; Cook, J.; Castonguay, P.; Popova, M.; Huang, J.; and Cohen, J. M. 2019. NeMo: a toolkit for building AI applications using Neural Modules. arXiv:1909.09577.

Lakhotia, K.; Polyak, A.; Hsu, W.-N.; et al. 2021. On Generative Spoken Language Modeling from Raw Audio. In *International conference on machine learning (ICML)*.

Lee, S.-g.; Ping, W.; Ginsburg, B.; Catanzaro, B.; and Yoon, S. 2023. BigVGAN: A Universal Neural Vocoder with Large-Scale Training. In *The Eleventh International Conference on Learning Representations*.

Maimon, G.; Roth, A.; and Adi, Y. 2025. Salmon: A suite for acoustic language model evaluation. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. IEEE.

McAuliffe, M.; Socolof, M.; Mihuc, S.; Wagner, M.; and Sonderegger, M. 2017. Montreal Forced Aligner: Trainable Text-Speech Alignment Using Kaldi. In *Proceedings of Interspeech*, 498–502.

Nguyen, T. A.; Muller, B.; Yu, B.; Costa-jussa, M. R.; Elbayad, M.; Popuri, S.; Ropers, C.; Duquenne, P.-A.; Algayres, R.; Mavlyutov, R.; Gat, I.; Williamson, M.; Synnaeve, G.; Pino, J.; Sagot, B.; and Dupoux, E. 2025. SpiRit-LM: Interleaved Spoken and Written Language Model. *Transactions of the Association for Computational Linguistics*, 13: 30–52.

Panayotov, V.; Chen, G.; Povey, D.; and Khudanpur, S. 2015. LibriSpeech: An ASR Corpus Based on Public Domain Audio Books. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5206–5210.

Radford, A.; Kim, J. W.; Xu, T.; Brockman, G.; McLeavey, C.; and Sutskever, I. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning (ICML)*, 28492–28518.

Reddy, C. K.; Dubey, H.; Gopal, V.; Cutler, R.; Braun, S.; Gamper, H.; Aichner, R.; and Srinivasan, S. 2021. ICASSP 2021 deep noise suppression challenge. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6623–6627. IEEE.

Saeki, T.; Xin, D.; Nakata, W.; Koriyama, T.; Takamichi, S.; and Saruwatari, H. 2022. Utmos: Utokyo-sarulab system for voicemos challenge 2022. *arXiv preprint arXiv:2204.02152*.

Tseng, L.-H.; Chen, Y.-C.; Lee, K.-Y.; Shiu, D.-S.; and Lee, H.-y. 2025. TASTE: Text-Aligned Speech Tokenization and Embedding for Spoken Language Modeling. *arXiv preprint arXiv:2504.07053*.

Wang, H.; Wang, H.; Guo, Y.; Li, Z.; Du, C.; Chen, X.; and Yu, K. 2024. Why Do Speech Language Models Fail to Generate Semantically Coherent Outputs? A Modality Evolving Perspective. *arXiv preprint arXiv:2412.17048*.

Yamagishi, J.; Veaux, C.; MacDonald, K.; et al. 2012. CSTR VCTK Corpus: English Multi-speaker Corpus for CSTR Voice Cloning Toolkit (version 0.80).

Yang, A.; Li, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Gao, C.; Huang, C.; Lv, C.; et al. 2025. Qwen3 technical report. *arXiv*.

Zen, H.; Dang, V.; Clark, R.; Zhang, Y.; Weiss, R. J.; Jia, Y.; Chen, Z.; and Wu, Y. 2019. LibriTTS: A Corpus Derived from LibriSpeech for Text-to-Speech. In *Proceedings of Interspeech 2019*, 1526–1530.

Zhang, X.; Zhang, D.; Li, S.; Zhou, Y.; and Qiu, X. 2024. SpeechTokenizer: Unified Speech Tokenizer for Speech Language Models. In *The Twelfth International Conference on Learning Representations*.

A Analysis of Results

A.1 Detailed Speech Reconstruction Results

To supplement the reconstruction results, we present reconstruction scores for the LibriTTS *test-clean* and *test-other* splits separately. This analysis provides a deeper understanding of the robustness of our SUBALIGN tokenization framework under varying acoustic conditions. Table 5 reports the reconstruction metrics. Notably, our framework, SUBALIGN, consistently achieves high perceptual quality and low word error rates on both the *test-clean* and *test-other* subsets, demonstrating its robustness even in noisy environments. Furthermore, SUBALIGN attains the lowest WER on the challenging *test-other* subset among all compared methods, further highlighting its effectiveness in more adverse acoustic conditions.

A.2 Qualitative Examples of Continuation

To provide deeper insight into our SUBALIGN-SLM 's continuation output, we include some of the continuation output and compare it with baseline outputs. Figure 4 shows the qualitative examples of speech continuation, where the non-bold segments are the input prompts and the bold segments are the continuations generated by each model. As shown, SUBALIGN-SLM generates more coherent and contextually appropriate continuations compared to the baselines. All texts are transcribed using `whisper-large-v3`.

A.3 Bitrate Calculation

The bitrate is calculated as the product of the bits per unit and the unit rate (i.e., the number of units per second). Formally:

$$\text{bitrate} = \text{bits per unit} \times \text{unit rate}. \tag{15}$$

In the case of SUBALIGN, the bits per unit are computed as:

$$\text{bits per unit} = \log_2 |\mathcal{V}_{\text{LLM}}| + R \cdot \log_2 K + \log_2 d_{\max}, \tag{16}$$

where $|\mathcal{V}_{\text{LLM}}|$ denotes the size of the LLM vocabulary, R is the number of RVQ stages, K is the codebook size, and $d_{\max}$ is the maximum duration. The unit rate is dynamic and aligned to the number of subword tokens per second, which varies depending on the input.

In the default configuration of SUBALIGN, we adopt the Llama vocabulary, with $|\mathcal{V}_{\text{LLM}}| = 128,256$, $R = 3$, $K = 512$, and $d_{\max} = 512$. Given these values, the bits per unit are approximately 52.97. As shown in Figure 3, the average unit rate on the LibriTTS *test-clean* subset is measured at 3.65 Hz, resulting in a bitrate of roughly 193 bps for SUBALIGN.

On the LibriSpeech *test-clean* subset, the average unit rate drops to 2.95 Hz, primarily because LibriSpeech transcripts lack punctuation, leading to a typically lower unit rate compared to LibriTTS. This results in a bitrate of about 156 bps on LibriSpeech.

While some previous works report unit rates and bitrates based on LibriSpeech (Baade, Peng, and Harwath 2025; Tseng et al. 2025), our model is typically trained and evaluated on transcripts that include punctuation marks. Therefore, we report the LibriTTS bitrate as the primary result for SUBALIGN.

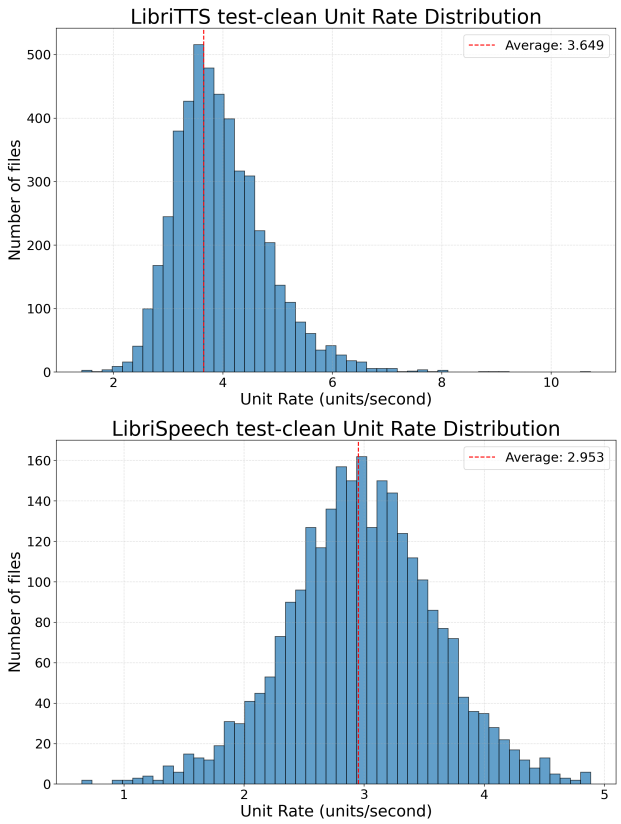

Figure 3: **Unit Rate Distribution for the LibriTTS (top) and LibriSpeech (bottom) *test-clean* subsets.** The average unit rate is calculated as the total number of units divided by the total duration. Since LibriSpeech transcripts do not contain punctuation marks, they generally exhibit a lower unit rate.

B Implementation Details

This section provides additional details on the training and inference procedures. To ensure reproducibility, we will make the code and model checkpoints publicly available.

B.1 Training Details

For the SUBALIGN tokenization module, we train the model for a total of 500K steps with a learning rate of 1e-4, weight decay of 1e-5, and a batch size of 8 distributed across 4 NVIDIA RTX A6000 GPUs. For SUBALIGN-SLM, we train for 200K steps, with the learning rate warmed up to 5e-5 in the first 100 steps, no weight decay, batch size of 128 across 4 NVIDIA RTX A6000 GPUs.

B.2 Inference Details

For speech continuation inference with SUBALIGN-SLM, we use separate configurations for predicting subword tokens and predicting acoustic tokens and durations. For subword tokens, we employ nucleus (top-p) sampling with a probability threshold of 0.3, a temperature of 0.5, and a repetition penalty of 1.3. For acoustic tokens and durations, we

Model	test-clean					test-other				
	WER ↓	UTMOS ↑	SECS ↑	F0 RMSE ↓	STOI ↑	WER ↓	UTMOS ↑	SECS ↑	F0 RMSE ↓	STOI ↑
Mel + BigVGAN	3.16	4.03	0.88	57.04	0.99	5.52	3.45	0.86	57.22	0.99
SpeechTokenizer - 4000	3.51	3.88	0.84	66.07	0.93	7.79	3.28	0.81	68.36	0.89
SpeechTokenizer - 2000	4.16	3.57	0.67	69.16	0.89	10.49	3.02	0.61	73.15	0.85
SpeechTokenizer - 1000	6.10	2.42	0.37	86.28	0.79	16.65	2.01	0.29	92.53	0.75
Mimi	4.40	3.63	0.64	65.09	0.91	10.73	3.11	0.62	68.52	0.88
WavTokenizer	4.96	3.97	0.73	67.35	0.92	11.25	3.43	0.70	68.77	0.89
S^3 Tokens	4.46	4.06	0.69	69.48	0.79	8.89	3.50	0.61	72.60	0.77
Sylber *(w/o quant.)*	5.34	4.09	0.40	69.12	0.76	12.85	3.89	0.34	75.67	0.72
TASTE	7.11	4.15	0.62	77.79	0.43	12.77	3.74	0.56	80.15	0.43
Ours *(w/o quant.)*	5.07	4.25	0.65	70.47	0.75	9.13	3.80	0.57	73.24	0.73
Ours	4.37	4.26	0.63	72.26	0.70	7.21	3.86	0.55	75.43	0.67

Table 5: **Comparison of reconstruction and perceptual metrics on test-clean and test-other.** We report WER, UTMOS, SECS, F0 RMSE, STOI for each split. Lower is better for ↓ metrics; higher is better for ↑ metrics. '*w/o quant.*' indicates models using continuous tokens.

Model	Prompt + Continuation
Cascaded (Llama)	We have a commander who's game for anything. **and he has a knack for it. He's been in combat more than any other player in the game, and he's one of the best. He's got the skills to pull it off.**
TWIST-7B	We have a commander who's game for anything. **We have a commander whose game, for anything we know, was marching along the Hudson and calling out to a party of men who lounged**
SpiRit-LM	We have a commander who's game for anything. **but fitting with the status quo or one and an offensive coordinator challenging players to show for passes but going the other way, and defensive coordinators doing a tougher job of preparation without a pass.**
TASLM (Llama)	We have a commander who's game for anything. **He's the best of the best. He's a very, very good commander.**
SUBALIGN-SLM (Llama)	We have a commander who's game for anything. **He's got a great attitude and is a very good leader.**
Model	**Prompt + Continuation**
Cascaded (Llama)	I met several of my friends. **and we had a nice time. I saw my friend, and we had a nice time.**
TWIST-7B	I met several of my friends. **I met several of my friends who are here.**
SpiRit-LM	I met several of my friends. **that it was coined to activist and together. Literally, it's many people are going to take the example of the academic boycotts for firsthand, as well as the festival. The mass have no immensely...**
TASLM (Llama)	I met several of my friends. **Hustle and surf for knowing the other party. Hats.**
SUBALIGN-SLM (Llama)	I met several of my friends. **We had a great time. I enjoyed the company of my friends.**

Figure 4: **Qualitative results on speech continuation.** Non-bold text indicates the prompt, while bold text represents the generated continuation.

use greedy decoding. Speech reconstruction similarly uses greedy decoding to predict S^3 tokens.

C Evaluation Details

This section outlines the evaluation process, including the guidelines given to human evaluators for assessing consistency of speech continuation, as well as the prompts used in GPT-4o-based evaluation of speech continuation. We use 100 audio samples from the LibriSpeech *test-clean* subset for both evaluation.

C.1 Human Evaluation Protocol

We assess how well the generated speech continuations remain consistent with the prompt speech via human listening tests, following the detailed instructions shown in Figure 5. Each test sample consists of a short audio Prompt (about 3 seconds) and a Prompt + Continuation (about 15 seconds). Raters evaluate the quality of only the continuation segment, i.e., the audio following the prompt, based on three criteria: prosody preservation, speaker similarity, and seamlessness of transition. For each sample, three independent reviewers assign an overall score from 1 (bad) to 5 (excellent) based on these predefined criteria.

Audio Continuation Evaluation Instructions

In this test, you will hear a short audio ‘‘Prompt’’ and a longer audio ‘‘Prompt + Continuation’’.
Please rate the **continuation** (the part after the prompt) on the following aspects:

- **Prosody Preservation:** Are the rhythm, intonation, and stress patterns preserved from the prompt into the continuation?
- **Speaker Similarity:** Does the continuation sound like it was spoken by the same person as the prompt?
- **Seamless Transition:** Is the point where the continuation begins smooth and hard to notice? (A higher score means a more seamless and natural transition.)

Scoring Definitions:

- **1 (Bad):** Prosody and voice are obviously different; clear, jarring change at the join point.
- **2 (Poor):** Noticeable difference in voice or rhythm; clear transition point.
- **3 (Fair):** Some similarities, but with noticeable change in voice or rhythm; transition point can be guessed.
- **4 (Good):** Mostly similar voice and prosody; only slight or subtle difference at the join; transition is hard to spot but not perfect.
- **5 (Excellent):** Completely seamless|same voice, same rhythm, no way to tell where the continuation starts; truly indistinguishable.

Figure 5: Human evaluation protocol for speech continuation: Raters judge prosody preservation, speaker similarity, and transition smoothness using a 5-point scale.

C.2 GPT-4o Prompt for Semantic Evaluation

For semantic relevance and coherence, we utilize GPT-4o (2024-08-06) as an automatic rater. The exact prompt provided to GPT-4o is shown in Figure 6. Following the protocol, GPT-4o rates each sample on a 5-point scale, considering both the contextual relevance and plausibility of the continuation for the given prompt.

Semantic Relevance Evaluation Prompt (GPT-4o)

```
[SYSTEM]
The task is to evaluate the relevance and likelihood of the predicted text
continuation, given a text prompt. You should also consider whether the meaning of the
text continuation makes sense.
The text prompt is: {prompt}
The text continuation is: {content}
Please provide an overall rating from 1 to 5 based on the following guideline:
1: The continuation is very unlikely and irrelevant to the prompt.
2: The continuation is unlikely and only marginally relevant.
3: The continuation is moderately likely and relevant.
4: The continuation is likely and relevant.
5: The continuation is very likely and highly relevant.
Please follow these steps:
First, briefly analyze the sample based on the above definitions.
Second, output the result in the following format:
I would rate the score as _
```

Figure 6: Prompt provided to GPT-4o for rating the semantic relevance and coherence of text continuations generated by speech models. This protocol is used to automatically assess the quality of generated speech continuations in our experiments.