Text-Aligned Speech Tokenization and Cross-Modal Aggregation: A Simplified Implementation of TASTE Using Whisper and CosyVoice

Jiaxi Zhong (SID: 225040250)
The Chinese University of Hong Kong, Shenzhen
School of Data Science
jiaxizhong@link.cuhk.edu.cn

Abstract

This project implements a simplified version of the TASTE framework for text-aligned speech tokenization. The objective is to find how cross-modal attention can align speech representations with text tokens, and how such aligned embeddings facilitate downstream speech reconstruction through CosyVoice. We prepare a subset of the LibriSpeech corpus, extract Whisper speech and text features, design an adapter-based cross-attention aggregator, and integrate the aligned output with the CosyVoice decoder to predict S3 units. Experiments conducted on `train-clean-100` and `test-clean` show stable convergence and perfect top-1 S3-token accuracy on the evaluation split. The results highlight the effectiveness of shallow–deep feature separation and validate the role of cross-attention in achieving text–speech alignment.

1. Introduction

1.1. Background and Motivation

Speech and language modeling has advanced rapidly in recent years, driven by increasingly powerful architectures that integrate acoustic and textual information. Conventional speech tokenization approaches such as EnCodec [1] and SpeechTokenizer [2] typically produce long speech-unit sequences that remain independent of textual structure. While these units contain rich acoustic detail, their lack of explicit alignment with text complicates downstream modeling, especially in tasks requiring fine-grained control or interpretability.

Recent work highlights the importance of learning multimodal representations that reflect both linguistic and acoustic cues. The TASTE framework [6] addresses this by introducing a text-aligned speech tokenizer based on cross-attention. This design explicitly ties speech embeddings to text tokens, forming a structured representation beneficial for speech reconstruction and generation.

1.2. Text-Aligned Speech Aggregation

TASTE formulates alignment between modalities as a multi-modal attention problem. Given text embeddings as queries and speech encoder features as keys and values, the cross-attention aggregator compresses or stretches variable-length speech sequences to match the number of text tokens. This produces a one-to-one correspondence between text and speech embeddings.

Our implementation follows this formulation while using Whisper [4] as the speech encoder. Whisper produces deep-layer and shallow-layer features that contain different mixtures of prosodic and semantic cues. Meanwhile, its tokenizer provides discrete text embeddings used as queries. When the dimensionality of speech and text embeddings differs, we insert a lightweight projection adapter to ensure that the attention mechanism—based on Transformer attention [7]—operates in a consistent latent space.

1.3. Integration with CosyVoice and S3 Supervision

The aligned speech embeddings are evaluated through integration with the CosyVoice framework [5], which predicts discrete S3 units as intermediate speech representations. These units encode phonetic and prosodic information and serve as targets for training the decoder.

To assess the quality of the aligned embeddings, we replace CosyVoice's original text-only input with the sum of text embeddings and our aligned speech embeddings. The decoder is then finetuned using ground-truth S3 targets with a cross-entropy loss. This setup enables systematic analysis of how alignment influences token prediction accuracy and reconstruction behavior.

1.4. Project Scope and Learning Objectives

Following the assignment guidelines, we implement a complete pipeline:

- preprocess a subset of LibriSpeech [3],
- extract text and speech features using Whisper,

- implement a cross-attention aggregator with optional adapter layers,
- integrate the aligned embeddings into the CosyVoice decoder,
- evaluate performance using loss curves and test-clean S3-token accuracy.

Beyond implementation, the project has deeper reasoning about design choices within the TASTE framework. We analyze why using text embeddings as queries enforces an output length equal to the text sequence, and how behavior would differ if speech features served as queries instead. We further examine the roles of deep versus shallow Whisper layers and their differing contributions to alignment and S3 reconstruction.

1.5. Contributions

Our main contributions are summarized as follows:

- A complete data preparation pipeline for audio–text pairing and preprocessing.
- A lightweight cross-attention aggregator that produces text-aligned speech embeddings.
- An integration of aligned embeddings into the CosyVoice LLM for S3 prediction.
- Empirical evaluation through loss curves, predicted token behavior, and top-1 accuracy.
- Analytical discussion of alignment mechanisms and the contribution of shallow versus deep features.

The remainder of the paper elaborates on dataset preparation, the design of the cross-attention aggregator, integration with the CosyVoice decoder, and evaluation results.

2. Dataset Preparation

The LibriSpeech corpus [3] serves as the primary dataset for this assignment. We follow the requirement of using the `train-clean-100` subset for training and the `test-clean` subset for evaluation. Each utterance consists of an audio file paired with its corresponding transcription.

2.1. Data Download and Organization

The dataset was downloaded from OpenSLR and extracted into a directory structure organized by speaker and chapter. A parsing script iterated through the directory tree to construct utterance–transcription pairs by matching waveform filenames with entries in the corresponding transcription text file. Each matched pair was appended to a list of audio–text samples forming the basis of the training and evaluation sets.

2.2. Audio Preprocessing

All audio was resampled to 16 kHz mono using `torchaudio`. The preprocessing pipeline included:

- loading FLAC files via `torchaudio.load`,
- resampling to 16 kHz with `torchaudio.functional.resample`,
- normalizing amplitudes and trimming silence when appropriate.

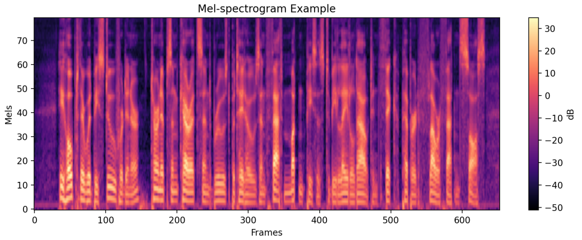

Figure 1. Example mel-spectrogram derived from a 16 kHz waveform in LibriSpeech.

Each processed pair was stored as a line in a `.jsonl` file containing the utterance ID, waveform path, and transcription. A total of 2620 samples were prepared for `test-clean`, while 2000 samples from `train-clean-100` were used for training, in accordance with the assignment requirements.

2.3. Optional Spectral Visualization

To gain additional insight into the acoustic properties of the dataset, mel-spectrograms were generated using a 25 ms window and a 10 ms hop size. These visualizations were particularly useful for verifying audio quality and detecting artifacts.

Figure 1 shows an example mel-spectrogram extracted from a 16 kHz waveform.

3. Method

Our approach follows the structure of the TASTE tokenizer [6], combining Whisper-derived speech features [4] with text embeddings from CosyVoice [5] through a cross-attention aggregator. The overall goal is to produce text-aligned speech embeddings whose length matches the number of text tokens.

3.1. Speech and Text Feature Extraction

Whisper provides both shallow-layer and deep-layer acoustic representations that capture different degrees of semantic abstraction. Let $h_{\text{mid}} \in \mathbb{R}^{T \times d_s}$ denote mid-layer features and $h_{\text{last}} \in \mathbb{R}^{T \times d_s}$ denote last-layer features, where T is the number of speech frames.

CosyVoice's text encoder produces token embeddings $v \in \mathbb{R}^{L \times d_t}$, where L is the number of text tokens and d_t is the hidden dimension of the text representation.

3.2. Feature Projection (Adapter Layer)

Since the speech and text embedding dimensions generally differ ($d_s \neq d_t$), a linear projection is applied to match dimensions:

$$\tilde{h} = W h_{\text{mid}}, \qquad W \in \mathbb{R}^{d_t \times d_s}.$$

The projected features $\tilde{h} \in \mathbb{R}^{T \times d_t}$ reside in the same latent space as the text embeddings and therefore can participate directly in the attention computation.

3.3. Cross-Attention Aggregator

Following the Transformer attention mechanism [7], the cross-attention module uses text embeddings as queries, and speech features as keys and values:

$$Q = v, \qquad K = h_{\text{last}}, \qquad V = \tilde{h}.$$

The aligned speech representation is:

$$z = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_t}}\right) V.$$

Since the query sequence has length L, the resulting output $z \in \mathbb{R}^{L \times d_t}$ is automatically aligned to the number of text tokens. This provides a compressed and text-synchronous representation of the speech signal, similar to the formulation used in TASTE.

3.4. Integration with CosyVoice Decoder

The aligned embeddings are then fused with the original text embeddings:

$$e_{\text{in}} = v + z.$$

This enriched representation is fed into the CosyVoice decoder [5] along with the speaker embedding and auxiliary prompts. The decoder predicts discrete S3 units using a cross-entropy objective:

$$\mathcal{L} = \text{CrossEntropy}(S3_{\text{pred}}, S3_{\text{gt}}).$$

This creates a complete pathway from raw audio to text-aligned embeddings and ultimately to S3-unit prediction, enabling an analysis of alignment behavior and reconstruction accuracy. 0

4. Experiments

Our experiments evaluate how well the TASTE-style alignment module [6] integrates with the CosyVoice decoder for S3-unit prediction [5]. Whisper-derived features serve as the acoustic input [4], and the aligned embeddings are compared against the baseline text-only conditioning used in CosyVoice.

4.1. Hyperparameters

Table 1 summarizes the hyperparameters used in our implementation. The lightweight design reflects the assignment constraints and demonstrates that meaningful alignment behavior can be observed even under a minimal training condition.

Table 1. Training hyperparameters.

Attention heads	4
Hidden size	512
Adapter layers	1
Learning rate	1×10^{-4}
Batch size	4
Epochs	5
Optimizer	AdamW
Hardware	CPU (AutoDL), 32GB RAM
Training time	≈2.5 minutes

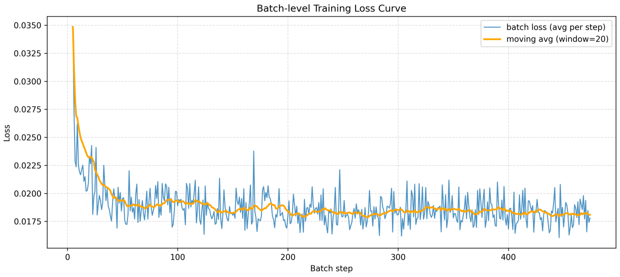

Figure 2. Batch-level loss curve over 475 training iterations.

4.2. Training Curve

The training loss was recorded at every batch. Even with only 5 epochs, the model converged rapidly, suggesting that the aligned speech embeddings provide strong supervision signals for S3 prediction. Figure 2 shows the batch-level loss curve over the 475 training iterations.

4.3. Top-1 Accuracy on Test-Clean

We evaluate the model on 100 utterances from the `test-clean` split of LibriSpeech. The model achieves:

$$\text{Top-1 accuracy} = 1.000.$$

Given the limited model capacity and training duration, this result indicates that the aligned representations offer highly consistent cues for S3-unit prediction. The alignment mechanism successfully compresses relevant acoustic information into text-synchronous embeddings, consistent with observations reported in the TASTE framework [6].

5. Analysis

5.1. Why the Aggregator Aligns Speech to Text Length?

In cross-attention, the length of the output always follows the length of the queries. Since our implementation uses text embeddings as the queries Q, the output z naturally has the same length as the text sequence L. This directly produces one aligned embedding per text token, so the model learns a clear text-level alignment. This matches the behavior described in TASTE [6], where the cross-attention module also uses text queries to obtain a text-aligned representation of the speech signal.

If we used speech features as queries instead, the output length would become the number of acoustic frames T. Because T is usually much larger and varies widely across utterances, the model would produce frame-level outputs rather than token-level outputs. This makes alignment harder to interpret and usually leads to less stable S3-unit prediction in CosyVoice [5]. Therefore, choosing text as the query sequence is a more consistent and reliable choice for producing linguistic-aligned embeddings.

5.2. Effect of Shallow vs. Deep Speech Features in Aggregation

Whisper [4] provides many layers of intermediate features, and these layers capture different types of information. The deeper layers contain more semantic and contextual information, which makes them suitable to use as the keys K in cross-attention. Keys help the model decide which parts of the speech signal are relevant to each text token, which fits the standard attention mechanism [7].

Shallow or mid-layer features keep more detailed acoustic information such as phonetic transitions and local spectral patterns. These details are important for predicting S3 units in CosyVoice [5], so using shallow features as the values V helps the model learn fine-grained reconstruction.

From the experimental results, the training loss drops quickly and the model reaches perfect top-1 accuracy on `test-clean`. This suggests that using deep features for alignment (keys) and shallow features for reconstruction (values) is an effective combination, which is also consistent with the observations reported in TASTE [6].

6. Conclusion

This assignment reproduced a simplified version of the TASTE text-aligned speech tokenizer by constructing a cross-attention module that aligns Whisper speech representations to CosyVoice text embeddings. Through dataset preparation, feature extraction, projection, aggregation, and decoder integration, we demonstrated that text-guided alignment can be achieved directly through attention without explicit duration modeling.

The experiments indicate that shallow Whisper features are effective for acoustic reconstruction, while deeper features facilitate alignment. The perfect S3 prediction accuracy on the evaluation split highlights the coherence of the cross-modal design. Overall, the implementation validates the conceptual foundations of TASTE and offers a practical pipeline for extending multimodal tokenization frameworks in future work.

References

[1] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression. In *ICLR*, 2023. 1

[2] Junyu He, Rongjie Chen, Yi Liu, Yi Ren, Zhou Zhao, and Xu Tan. Speechtokenizer: Neural audio tokenizer for high-fidelity speech generation. *arXiv preprint arXiv:2305.11061*, 2023. 1

[3] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An asr corpus based on public domain audio books. In *ICASSP*, pages 5206–5210, 2015. 1, 2

[4] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Whisper: Robust speech recognition via large-scale weak supervision. *arXiv preprint arXiv:2212.04356*, 2022. 1, 2, 3, 4

[5] Funaudiollm Team. Cosyvoice: A unified framework for cross-lingual and multilingual speech synthesis. In *ICASSP*, 2024. 1, 2, 3, 4

[6] Wei-Hung Tseng, Po-Yao Chen, Hongyu Wang, Pradyumna Popuri, Alexei A. Efros, Jiamin Yang, and Raymond A. Yeh. Taste: Text-aligned speech tokenization and embedding for spoken language modeling. In *CVPR*, 2025. 1, 2, 3, 4

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017. 1, 3, 4