Balancing Speech Understanding and Generation Using Continual Pre-training for Codec-based Speech LLM

Anonymous ACL submission

Abstract

Recent efforts have extended textual LLMs to the speech domain, yet a key challenge remains: balancing speech understanding and generation while avoiding catastrophic forgetting when integrating acoustically rich codec-based representations into models originally trained on text. In this work, we propose a novel approach that leverages continual pre-training (CPT) on a pre-trained textual LLM to create a codec-based speech language model. This strategy mitigates the modality gap between text and speech, preserving the linguistic reasoning of the original model while enabling high-fidelity speech synthesis. We validate our approach with extensive experiments across multiple tasks—including automatic speech recognition, text-to-speech, speech-to-text translation, and speech-to-speech translation (S2ST)—demonstrating that our model achieves superior TTS performance and, notably, the first end-to-end S2ST system based on neural codecs.

1 Introduction

The textual large language model (LLM) has significantly influenced the natural language processing community, achieving exceptional performance across a variety of tasks, both in-domain and out-of-domain, with consistent reliability (Zhao et al., 2023; Achiam et al., 2023; Dubey et al., 2024; Yang et al., 2024a). A central element of its success is the auto-regressive architecture, which can be abstracted to handle a wide range of NLP tasks in a unified form. This approach not only boosts the model's versatility but also positions it as a foundational model for diverse downstream applications.

Building on the robust capabilities of textual LLMs, recent work has introduced several examples of speech LLMs. Some exhibit strong performance in spoken language understanding tasks (Kharitonov et al., 2022; Gong et al., 2023, 2024; Chang et al., 2024a; Tang et al., 2024a; Huang et al., 2024b; Dubey et al., 2024; Rubenstein et al., 2023; Kuan et al., 2024; Zhang et al., 2023a; Maiti et al., 2024), while others demonstrate reasonable generalization in various speech generation tasks (Wang et al., 2023; Anastassiou et al., 2024; Kim et al., 2024; Défossez et al., 2024a; Chen et al., 2024a; Tian et al., 2024; Wang et al., 2024; Huang et al., 2024c; Yang et al., 2024b).

A key challenge in speech LLMs is balancing speech understanding and generation while mitigating catastrophic forgetting when incorporating speech modalities into a textual LLM. Codec-based representations offer high-fidelity speech synthesis by preserving fine-grained acoustic details (Wang et al., 2023; Yang et al., 2024b; Défossez et al., 2024a; Kim et al., 2024) but exhibit limitations in understanding tasks due to their primary focus on resynthesis (Shi et al., 2024; Chang et al., 2024c; Dhawan et al., 2024). Moreover, integrating codec tokens into a textual LLM introduces a substantial modality gap, as textual models are not inherently designed to process acoustically rich representations. This shift in knowledge distribution can disrupt the model's linguistic reasoning, leading to catastrophic forgetting of previously learned text-based capabilities. While codec-based models have shown promise in speech generation, their effective integration into a unified speech LLM remains an open problem.

To address the above challenge, this paper proposes using continual pre-training (CPT) on a pre-trained textual LLM to realize a *codec*-based speech language model for the first time. The CPT helps mitigate the catastrophic forgetting from modality mismatch and connect speech codec sequences to the language knowledge embedded in the original textual LLM. Extensive experiments are conducted over the continual pre-trained codec speech LLM with further supervised fine-tuning (SFT) across various tasks, including automatic speech recognition (ASR), text-

to-speech (TTS), speech-to-text translation (S2TT), and speech-to-speech translation (S2ST). The results demonstrate that continual pre-trained speech LLM on speech-only data can achieve significantly better TTS performance. Notably, we show that the model can *perform well in S2ST in a single end-to-end model based on neural codecs*, which can only be achieved with semantic-oriented discrete speech representation in previous works (Lee et al., 2022). Our contribution includes:

- first proposing CPT to mitigate the catastrophic forgetting from mismatched modalities for speech codecs-based LLM.
- designing a practical solution for the CPT and SFT with the speech codec-based LLM.
- utilizing the CPT-ed model to a range of speech tasks, which shows comparable or better performance to specialized models in different tasks. Notably, we achieved the first successful S2ST system in a single end-to-end model based on neural codecs.

2 Related Works

2.1 Speech Language Modeling

Discrete tokens offer a natural connection to textual LLMs. SSL-based discrete tokens can seamlessly integrate with textual LLMs, functioning similarly to text tokens (Rubenstein et al., 2023; Maiti et al., 2024; Wu et al., 2024c; Zhang et al., 2023b). A common practice in these works is to use a combined vocabulary for both speech tokens and textual tokens, enabling the model to perform both understanding and generation tasks. However, a vocoder is required to enable the speech generation. The quality of generation is often limited by the lossy nature of SSL representations and the additional clustering process involved (Lee et al., 2022; Shi et al., 2023; Tang et al., 2024b), especially in multi-speaker setups where previous works need to utilize additional speaker embedding to stabilize the training (Maiti et al., 2024; Guo et al., 2025).

To improve generation quality, codec tokens can be used in speech LLMs (Wang et al., 2023; Anastassiou et al., 2024; Kim et al., 2024; Défossez et al., 2024a; Chen et al., 2024a; Tian et al., 2024; Wang et al., 2024; Yang et al., 2024b). Unlike SSL-based tokens, codec tokens often require multiple streams or levels of code to achieve high-quality resynthesis. By balancing efficiency, performance, and interdependencies among different codec streams, various interleaving patterns can be employed to predict multi-stream codecs (Wang et al., 2023; Copet et al., 2023; Yang et al., 2024b). Using codec tokens, speech synthesis quality can be significantly enhanced. However, empirical analysis suggests that codec tokens may have limited understanding capabilities when applied to train an ASR system, particularly in comparison to SSL-based tokens (Shi et al., 2024; Chang et al., 2024b; Dhawan et al., 2024).

Based on the above literature review, the primary challenge in utilizing SSL-related features arises from their relatively lossy conversion, which highlights a performance bottleneck at the vocoder stage during speech reconstruction. This suggests that a standalone speech LLM relying on SSL-based tokens is unlikely to resolve this issue. Instead, we use the other type of speech tokenizer, speech codecs in our main framework. In contrast, while codec-based speech LLMs may exhibit limitations in understanding tasks, their high resynthesis quality ensures effective information transfer within the model, offering greater potential for speech LLM development. Therefore, we have chosen codec-based speech LLMs as the focus for our CPT efforts.

2.2 Continual Pre-training for Speech Codec-based LLM

Continual pre-training has proven to be an effective strategy in textual LLMs, particularly when there is a significant shift or expansion in the model's knowledge base (Wu et al., 2024b). While speech signals have shared semantic information with text, speech also contains diverse additional information, including paralinguistics, and environments. Though both speech and text are in sequences, speech signals tend to be longer due to richer information (Chen et al., 2022; Wang et al., 2022; Han et al., 2021). Due to these differences from the modality mismatch, the introduction of speech modality to a textual LLM can significantly alter the knowledge base of an originally pre-trained textual LLM, causing *catastrophic forgetting* to the model. As a result, several previous works have shown that the combination of speech-related modules and textual LLMs can result in *catastrophic forgetting* on common natural language understanding tasks and speech processing tasks (Zhan et al., 2024; Huang et al., 2024b,a; Chu et al., 2024b). Continual pre-training, which involves an additional pre-training stage, is used to extend the

general knowledge of a pre-trained model without focusing on task details. Previous works have applied CPT in speech LLMs with SSL tokens by expanding the vocabulary of pre-trained textual LLMs (Zhang et al., 2023a; Rubenstein et al., 2023), demonstrating its effectiveness in integrating speech components into textual LLMs. But SSL-based tokens inherently struggle to retain sufficient acoustic information for generation tasks.

Neural codecs, on the other hand, provide higher quality in speech reconstruction by capturing more detailed acoustic information, approaching the fidelity of raw speech signals (Wu et al., 2024a; Shi et al., 2024; Kim and Skoglund, 2024). Despite this, there is a substantial domain gap between textual and codec tokens due to the codec's emphasis on acoustic details. Additionally, codec tokens are often represented in multi-stream setups to achieve high-fidelity acoustic details, necessitating modifications to the model architecture to bridge the two modalities, which introduces further complexity in model changes (Chen et al., 2024a; Wang et al., 2023; Copet et al., 2023; Défossez et al., 2024b).

Given these challenges, we propose employing CPT to align codec tokens with textual tokens. This approach aims to mitigate catastrophic forgetting while effectively linking speech codec sequences to the linguistic knowledge embedded in the original textual LLM. By doing so, we enhance the model's capability in both speech understanding and generation tasks. Our method capitalizes on the strong comprehension abilities of pre-trained textual LLMs while leveraging the superior generation quality of codec-based speech LLMs, creating a more robust and integrated multimodal system.

3 Methodology

This section outlines the key components of the proposed method, including the speech tokenizer, model architecture, and the training strategy employed for CPT. The first two subsections (i.e., speech tokenizer and model architecture) provide a practical base framework for CPT, where the third section discusses the novel CPT strategy applied to the model training.

3.1 Speech Tokenizer

As discussed in Sec. 2, we utilize neural codecs as the speech tokenizer in this work. The speech tokenizer, denoted as $\mathrm{Tokenizer}(\cdot)$, consists of an $\mathrm{Encoder}(\cdot)$, a quantizer $\mathrm{Quantizer}(\cdot)$, and a decoder $\mathrm{Decoder}(\cdot)$. The quantizer includes a set of L codebooks, where the i^{th} codebook, $\mathcal{B}^i = \{b_1^i, b_2^i, ..., b_{B^i}^i\}$, contains B^i codes.

Given a sampled speech signal $S \in \mathbb{R}^{1\times T_S}$ with length T_S, the encoder processes the signal into hidden states $Q = \mathrm{Encoder}(S) \in \mathbb{R}^{D^{\text{embed}}\times T_Q}$, where D^{embed} represents the dimension of each frame in the hidden states, and T_Q is the temporal sequence length. The quantizer then transforms Q into discrete codes $C \in (\mathcal{B}^1 \times \mathcal{B}^2 \times ... \times \mathcal{B}^L)^{T_C}$ across T_C frames:

$$C, E = \mathrm{Quantizer}(Q|\mathcal{B}^1, \mathcal{B}^2, ..., \mathcal{B}^L), \quad (1)$$

where the hidden states $E \in \mathbb{R}^{D^{\text{embed}}\times T_C}$ is constructed from the discrete codes C. Specifically, using the codebooks, the discrete code C is firstly mapped into corresponding embeddings $M \in \mathbb{R}^{L\times D^{\text{embed}}\times T_C}$. M are then compressed along the codebooks' axis to form E. To reconstruct the speech signal S, the decoder takes E as input and generates the reconstructed signal $\hat{S} = \mathrm{Decoder}(E)$. The discrete codes C are used as the I/O of speech signals for codec-based speech LLM.

3.2 Model Architecture

The model architecture is illustrated in Figure 1. In the first step, both speech codec tokens and textual tokens are transformed into multi-modal shared embeddings. For codec tokens, we select a subset of $L'(\leq L)$ streams of codes, denoted as $C' \in (\mathcal{B}^1 \times \mathcal{B}^2 \times ... \times \mathcal{B}^{L'})^{T_C}$, from the full set of codec tokens C in the input speech stream.[1] Each code in C' is mapped into corresponding embeddings $M' \in \mathbb{R}^{L'\times D^{\text{embed}}\times T_C}$ and is then aggregated into $E^{\text{speech}} \in \mathbb{R}^{D^{\text{embed}}\times T_C}$. Textual tokens $A \in \mathcal{V}^{T_A}$ with a vocabulary of $\mathcal{V}$ are directly transformed into embeddings $E^{\text{text}} \in \mathbb{R}^{D^{\text{embed}}\times T_A}$ with a sequence length of T_A. Both E^{speech} and E^{text} (denoted as E) are used seamlessly within the LLM's decoder-only architecture. For simplicity, we denote $R \in (\mathcal{B}^1 \times \mathcal{B}^2 \times ... \times \mathcal{B}^{L'} \times \mathcal{V})^{T_R}$ that represents both speech codec tokens and text tokens. The T_R can be the length of either a speech codec token sequence (i.e., T_C) or a text token sequence (i.e., T_A).

The core of the LLM is adapted from Qwen1.5, a Transformer-based LLM (Yang et al., 2024a).[2]

[1] L' is a hyperparameter chosen to balance computational efficiency, modeling complexity in LLM training, and audio reconstruction quality.

[2] While Qwen1.5 has demonstrated superior performance compared to other recent open-source textual LLMs (Yang et al., 2024a), it also offers a 0.5B version that fits within our

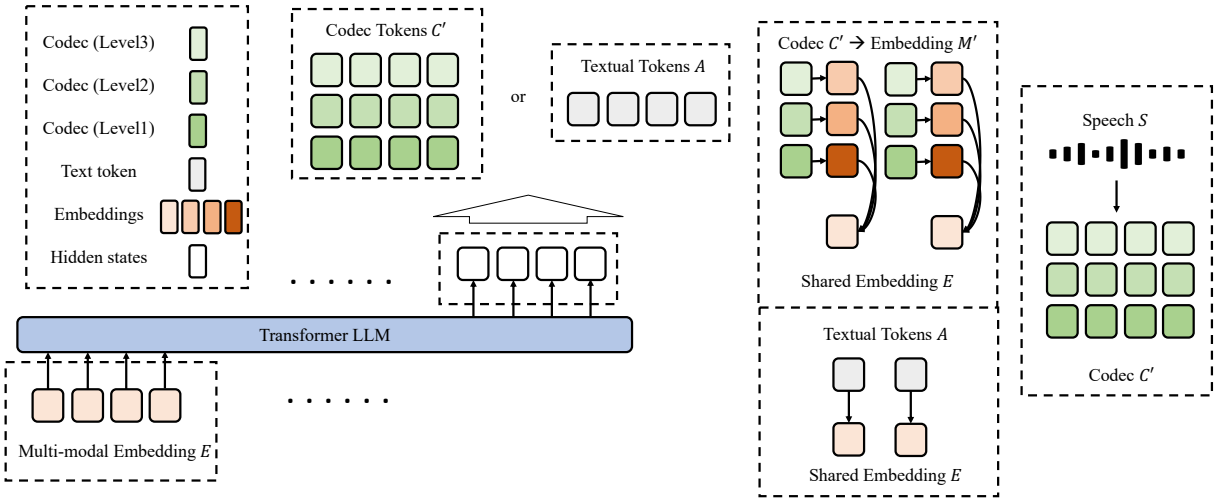

Figure 1: The architecture of the codec-based speech LLM. Codec tokens and textual tokens are converted into multi-modal shared embeddings, as shown on the right side of the figure. These shared embeddings are then fed into a Transformer-based LLM, which features parallel prediction heads designed for predicting either codec tokens or textual tokens. Details are discussed in Sec. 3.2.

To retain the majority of pre-trained knowledge for CPT, we preserve the original Transformer architecture along with the same textual tokenizer and textual token embeddings. Given the hidden states from the final Transformer layer, the language model prediction head is partially initialized with the pre-trained Qwen1.5 weights, while the portion related to the extended vocabulary for codec tokens is initialized randomly. Additional language prediction heads are introduced to predict subsequent codec levels *in parallel*. Unlike recent works (Chen et al., 2024a; Copet et al., 2023; Yang et al., 2024b), we do not enforce dependency constraints between codec predictions. Compared to approaches that use non-autoregressive networks (Wang et al., 2023) or multi-scale Transformers (Yang et al., 2024b), our parallel prediction style remains simple yet efficient during inference.

3.3 Model Training

The formulation discussed in the previous two sections provides a unified interface for speech and text modalities within a single end-to-end model, where speech is represented in multiple streams and text in a single stream. Notably, this is the first approach to seamlessly integrate multi-stream speech codecs with text tokens.

However, this formulation has limitations when incorporating a pre-trained textual LLM due to significant modality mismatches, particularly with speech codecs. To address this, we propose using CPT to align modalities, enabling a unified model for both understanding and generation. This approach leverages the strong foundation of textual LLMs while maintaining high-fidelity speech acoustics through advanced codecs.

In the original textual LLM, the pre-training objective is defined as:

$$\theta_{\text{PT-}A} = \text{argmax}_\theta \prod_t^{T_A} P_\theta(a_t|a_0, a_1, ..., a_{t-1}), \tag{2}$$

where θ is the language model parameters. When applying CPT using only speech data, the objective transforms into:

$$\theta_{\text{CPT-}C} = \text{argmax}_\theta \prod_t^{T_C} P_\theta(c_t|c_0, c_1, ..., c_{t-1}). \tag{3}$$

The purpose of CPT is to inject knowledge about speech modality into the LLM transformer base. Therefore, we conduct two pre-training configurations: one with speech data only following Eq. (3), and the other with bi-modal (i.e., speech and text) datasets. Details of data preparation are discussed in Sec. 4.1. We denote a sequence of speech and text tokens $R = (r_1, r_2, ..., r_{T_R})$ with a length of T_R. $r_t \in (\mathcal{B}^1 \times \mathcal{B}^2 \times ... \times \mathcal{B}^{L'} \times \mathcal{V})$ can represent either a speech codec token $(c_t^1, c_t^2, ..., c_t^{L'}, \text{null})$ or text token $(\text{null}, ..., \text{null}, a_t)$. The pre-training objective follows the basic next-token prediction task:

$$\theta_{\text{CPT}} = \text{argmax}_\theta \prod_t^{T_R} P_\theta(r_t|r_0, r_1, ..., r_{t-1}), \tag{4}$$

computational constraints. Therefore, in this work, we focus on Qwen1.5 as our base textual LLM.

where θ is the model illustrated in Fig. 1 and r_0 is the start of sentence.

Following the CPT stage, we conduct SFT where a prompt token sequence R^{inp} is first processed before the target sequence R^{tgt}. The objective for the SFT is:

$$\theta_{\text{SFT}} = \text{argmax}_\theta P_\theta(R^{\text{tgt}}|R^{\text{inp}}). \quad (5)$$

4 Experimental Setup

In this section, we detail the experimental settings for CPT and SFT to evaluate the effectiveness of the proposed method across various downstream tasks. Specifically, we compare CPT-ed models with those that are either randomly initialized or initialized from a textual LLM.

The basic speech tokenizer follows the SoundStream architecture (Zeghidour et al., 2021). Instead of using a complex short-term Fourier transform (STFT) discriminator, we adopt the discriminators from (Kumar et al., 2023), which include a multi-frequency STFT discriminator, a multi-scale discriminator, and a multi-period discriminator. For more detailed hyperparameters of the speech tokenizer, please refer to Appendix A.

4.1 Pre-training Data Preparation

As mentioned in Sec. 1, we focus on codec-based speech LLMs. Therefore, in the following experiments, we primarily consider bilingual scenarios, using both English and Mandarin data.

The pre-training data consists of around 140k hours of English and Mandarin speech, along with corresponding transcriptions and translations. Given the limited availability of open-source Mandarin speech data, we incorporate 70k hours of in-house Mandarin data to maintain a balance between English and Mandarin in the pre-training corpus. More detailed information about the corpora used for pre-training is provided in Appendix B.1. For training consistency on punctuation, we use a BERT-based punctuation restoration model[3] to recover the punctuation for textual data without punctuation (e.g., Librispeech (Panayotov et al., 2015)).

Most of the corpora included were originally designed for ASR or TTS purposes and did not include translations in their official release, which may prevent full alignment of the semantic spaces between the two languages. To semantically align English and Mandarin in the speech LLM, we supplement the data using an internal machine translation model to generate translations between English and Mandarin (EN->ZH and ZH->EN) based on the original speech transcriptions.[4] For each utterance in the pre-training dataset, we use the paired tuple (i.e., speech, transcription, and translation) to create six tasks: (1) speech continuation, which predicts future speech tokens based on previous segments; (2) language modeling, which generates the following textual tokens from a given context; (3) ASR, which transcribes speech into text; (4) TTS, which generates speech from text using a target speaker's speech segment as a prompt; (5) speech-to-text translation (S2TT), which translates speech in one language to text in another; and (6) text-to-speech translation (T2ST), which generates translated speech from a source text using a target speaker's prompt.

We did not include S2ST during continual pre-training because (1) our ASR/TTS corpora lack native S2ST pairs, making high-quality paired data scarce, and (2) synthesizing speech that fully preserves speaker identity and natural prosody is challenging. Instead, we reserve S2ST as an unseen task and later verify its performance via supervised fine-tuning, allowing us to assess whether our CPT framework has successfully aligned the English and Mandarin semantic spaces.

Based on the six tasks mentioned, the final sequence is formulated using the pattern "<*Condition*><*Prompt*><*Target*>". Additionally, a boundary token is inserted for each speech or text segment to mark the start or end of the corresponding segment. Depending on the specific task type, we enforce slightly different policies generating the data. The specifics are detailed in Appendix B.2. Noted that the natural language prompts are only in textual modality.

4.2 Continual Pre-training

We adopt `Megatron-LM`[5] with tensor parallelism as our training framework (Shoeybi et al., 2019), using Qwen1.5-0.5B as the base pre-trained textual LLM (Yang et al., 2024a).

As formulated in Eq. (4), we carry out two CPT experiments: one using only speech data, and the other incorporating both speech and text modalities.

[3] https://huggingface.co/felflare/bert-restore-punctuation

[4] We use our internal system because it's fine-tuned on our speech corpora and integrated with our data pipeline, ensuring more consistent and accurate translations for our tasks.

[5] https://github.com/NVIDIA/Megatron-LM

For the mixed-modality training, we aim to balance different domains and tasks by constructing a data loader that samples data from various tasks. Specifically, the loader randomly selects data for each of the six speech-related tasks with a 15% probability. To maintain the reasoning capacity of the original textual LLM and avoid catastrophic forgetting, we include text-only data for the remaining 10%. Of this textual data, 5% comes from general domains such as books, YouTube titles, and Wikipedia, while the other 5% is sourced from an in-house machine translation (MT) corpus to enhance the model's translation capabilities. This design aligns with previous CPT approaches in both textual and multimodal LLMs, where even limited text-based pre-training acts as a stabilizing regularizer, anchoring the model's internal representations to those learned during pre-training (Zhai et al., 2023; Sun et al., 2020). Detailed training hyperparameters are discussed in Appendix C.

Additionally, to identify the effect of CPT, we conduct experiments with a random-initialized 0.5B model that has the same architecture as Qwen1.5-0.5B and the pre-trained Qwen1.5-0.5B for comparison.[6]

4.3 Downstrem Tasks

As formulated in Eq. (5), the downstream evaluation is performed by fine-tuning the pre-trained model across four tasks: ASR, TTS, S2TT, and S2ST. ASR, TTS, and S2TT are used to assess the model's basic speech understanding and generation abilities. S2ST can be broken down into ASR, MT, and TTS, making it a more comprehensive task that evaluates both understanding and generation capabilities.

All four fine-tuning tasks follow the sequence formulation discussed in Sec. 4.1, with different <*Condition*>, <*Prompt*>, and <*Target*>, related to the tasks. Notably, the task prompts are also generated following the pipeline in Sec. 4.1 but they do not overlap with the pre-training prompts.

ASR: For the ASR task, we evaluate the model using the Librispeech dataset for English and the Aishell2 dataset for Mandarin (Panayotov et al., 2015; Du et al., 2018). The <*Condition*> consists of a sequence of speech codec tokens, followed by a natural language prompt. As described in Sec. 4.1, the <*Target*> sequence is the transcription with restored punctuation. To further improve model performance, we apply random time-domain masking, similar to the approach in (Chang et al., 2023, 2024c).

During inference, the model autoregressively predicts the transcription by feeding the <*Condition*><*Prompt*> sequence into the system. To enhance decoding performance, we employ beam search, using a beam size of 8. We measure word error rate (WER) for English ASR and character error rate (CER) for Mandarin ASR.

TTS: We focus on the multi-speaker TTS task with LibriTTS (Zen et al., 2019). For the task formulation, we follow VaLL-E style input where the condition includes a three-second speaker prompt and the text (Wang et al., 2023). For the target text, we restore the punctuation similar to the ASR task.

Since greedy search tends to produce trivial outputs, we adopt a sampling-based inference with a top-k strategy, setting $k = 30$, as used in prior works (Wang et al., 2023; Yang et al., 2024b; Tian et al., 2024). To further increase the diversity of generated speech, we re-scale the predicted logits using a temperature of 1.5.

For evaluation, we use: WER from a pre-trained ASR model, speaker similarity (SPK-SIM) from a pre-trained speaker embedding model, and an automatic speech quality predictor based on a pre-trained mean opinion score (MOS) predictor. Specifically, we use Whisper-large-V3 (Radford et al., 2023) for WER evaluation, a pre-trained Rawnet3 model (Jung et al., 2024) trained on Voxceleb for speaker embedding extraction, and UTMOS (Saeki et al., 2022) as the MOS predictor. Following common practice in previous works (Wang et al., 2023; Yang et al., 2024b; He et al., 2024; Tian et al., 2024), we generate five samples per test instance using the sampling strategy and report the average score across each metric.

S2TT: The S2TT adopts a task formulation as the ASR task by simply replacing the input speech with source language speech and target transcription with target language translation. The prompts are changed to task-related prompts accordingly. We test two corpora: CoVOST2 and GigaST (Wang et al., 2021; Ye et al., 2023). For CoVOST2, we focus on two translation directions, including English-to-Mandarin (EN->ZH) and Mandarin-to-English (ZH->EN). For GigaST, we only focus on EN->ZH. We use SacreBLEU to evaluate the prediction results with BLEU score (Post, 2018).

S2ST: We conduct S2ST using the GigaS2S corpus, which supplements the GigaST corpus with a

[6]They are denoted as "No Initialization" and "Text LLM Initialization" in the following discussion.

Table 1: ASR performance on LibriSpeech and Aishell2. Models marked with * indicate pre-trained models did not undergo continual pre-training. + stands models that do not use neural codecs as their speech representation. We report WER for Librispeech and CER for Aishell2.

Models	Param.	LibriSpeech		Aishell2
		Test-clean	Test-other	Test-overall
VoxtLM+ (Maiti et al., 2024)	1B	**2.7**	6.5	-
AnyGPT (Zhan et al., 2024)	7B	8.5	-	-
No Initialization*	1B	5.5	9.5	15.5
Text LLM Initialization*		4.8	8.5	13.1
Speech CPT		5.5	8.9	13.0
Speech & Text CPT		3.7	**6.3**	**7.2**

single-speaker TTS model (Ye et al., 2023).[7] Due to data constraints, we focus only on the English-to-Mandarin (EN->ZH) translation direction. Since the target speech is single-speaker, the <*Condition*> consists solely of source language speech (i.e., English). The prompt generation process follows the same approach as in previous tasks. Consistent with the method used in AudioPalm (Rubenstein et al., 2023), we directly assign the <*Target*> as the codec sequence of the target speech.

For inference, we use the same top-k sampling strategy employed in the TTS task. For evaluation, we measure translation quality using ASR-BLEU and speech quality using UTMOS.[8]

5 Results and Discussion

5.1 ASR

The experimental results for ASR are presented in Table 1. The best performance is achieved by the model that underwent CPT with both speech and text modalities. This outcome is expected since ASR is one of the tasks included in the pre-training phase. The model that received only speech CPT performed worse on Librispeech, indicating that focusing solely on speech continuation within the speech codec pre-training does not necessarily enhance speech recognition performance.

We also compare these results with reference performances from other speech LLM-based models on Librispeech. VoxtLM, which uses SSL-based tokens as its modeling unit (Maiti et al., 2024), and AnyGPT, a multi-modal LLM that uses SpeechTokenizer—a speech neural codec that additionally distills representations from speech self-supervised models (Zhan et al., 2024; Zhang et al., 2024)—are included in the comparison. While the proposed model with CPT slightly degraded from VoxtLM on the Librispeech text-clean set, it outperforms on the more realistic test-other set.[9] This potentially suggests that CPT successfully improves the model's understanding ability in codec-based speech LLMs, bringing its performance closer to that of SSL representations, which are known to excel in understanding tasks.

Table 2: TTS performance on LibriTTS. Models marked with * indicate pre-trained models that did not undergo continual pre-training. ° corresponds to a version trained on LibriTTS.

Models	UTMOS	WER	SPK-SIM
UniAudio° (Yang et al., 2024b)	3.64	13.1	0.43
No Initialization*	3.01	17.5	0.55
Text LLM Initialization*	2.78	18.8	0.51
Speech CPT	**3.65**	**3.7**	**0.66**
Speech & Text CPT	3.59	**3.7**	0.65

5.2 TTS

The TTS performance results are presented in Table 2. The best-performing system is the model that uses speech-only CPT, indicating that speech continuation pre-training can significantly enhance speech generation quality and ease the challenges associated with speech generation modeling. The model with joint speech and text CPT achieves comparable performance in terms of intelligibility, as measured by WER, and in speaker prompt understanding, as indicated by SPK-SIM. Overall, models that apply CPT have demonstrated superior performance compared to those with or without textual LLM initialization.

As shown in Table 2, We also conduct experiments on the LibriTTS dataset using the UniAudio-based model (i.e., multi-scale transformer-based language model TTS model) in ESPnet (Shi et al., 2024; Tian et al., 2024; Yang et al., 2024b). The same speech tokenizer as our CPT-ed model is employed for these experiments. Compared to the TTS-specialized model, the results demonstrate that the proposed method generates speech with comparable quality, as measured by UTMOS, while significantly improving intelligibility and speaker style transfer, as reflected by much lower WER and higher SPK-SIM scores.

[7] https://github.com/SpeechTranslation/GigaS2S

[8] Although UTMOS was trained on English speech, which might introduce some language mismatch in the scoring, prior work (Huang et al., 2022) has shown that UTMOS still achieved reasonable correlation scores when evaluating out-of-domain Chinese speech. Thus, we continue to use it for our speech quality evaluation.

[9] Due to differences in pre-training data and model size, concrete comparisons are challenging.

Table 3: S2TT performance on CoVOST2 and GigaST. The performance is reported in BLEU. Models marked with * indicate pre-trained models that did not undergo continual pre-training. • stands that external machine translation data is used.

Pre-training	CoVoST2		GigaST
	EN -> ZH	ZH -> EN	EN -> ZH
Fairseq ST (Wang et al., 2021)	25.4	5.8	-
OWSM-v3 (Peng et al., 2023)	**33.4**	13.6	-
GigaST• (Ye et al., 2023)	-	-	38.0
LLM-ST• (Huang et al., 2023)	-	-	**39.6**
No Initialization*	25.5	5.8	30.4
Text LLM Initialization*	28.9	9.9	33.2
Speech CPT	24.8	5.4	33.1
Speech & Text CPT	33.1	**16.1**	37.5

Combining these results with those from Sec. 5.1, our proposed method not only enhances the ASR performance of codec-based speech LLMs but also maintains high TTS quality, ensuring no degradation in speech generation compared to other specialized codec-based speech language models without textual LLM initialization.

5.3 S2TT

The results for speech-to-text translation are shown in Table 3. In both the CoVOST2 and GigaST datasets, models with CPT using both speech and text modalities demonstrate significant improvements in the S2TT task, highlighting their effectiveness in understanding tasks. Notably, the model with speech-only CPT performs worse than models without initialization. This result aligns with the ASR findings but contrasts with the TTS results, indicating that a focus on speech generation does not necessarily enhance speech understanding ability.

We also present results from related works (Wang et al., 2021; Peng et al., 2023; Ye et al., 2023; Huang et al., 2023). Our proposed model shows better performance than Fairseq-ST and achieves performance comparable to OWSM-v3, which focuses on understanding tasks. For GigaST, since both models incorporate external machine translation data, a direct comparison is not possible. But we observe that the model with joint-modality CPT has achieved performance similar to these models that are specifically designed for S2TT tasks.

5.4 S2ST

The results for S2ST are presented in Table 4. The best performance is achieved by the model with joint speech-text CPT. Notably, unlike ASR, TTS, and S2TT, S2ST is not included as a task in the CPT phase. However, both models that underwent CPT still demonstrated strong S2ST performance. In contrast, models without CPT, even after extensive hyper-parameter tuning, struggled to converge effectively on the S2ST task.[10]

Table 4: S2ST performance on GigaST. † indicates that the ASR-BLEU scores were calculated using different ASR systems, as described in Sec. 5.4.

Pre-training	GigaST	
	ASR-BLEU	UTMOS
Vec-Tok† (Zhu et al., 2023)	21.6	-
HW-TSC† (Wu et al., 2024d)	**33.6**	-
Speech CPT	28.0	3.41
Speech & Text CPT	33.4	**3.66**

For comparison, we also include results from two prior works (Zhu et al., 2023; Wu et al., 2024d). Vec-Tok uses an end-to-end architecture with additional emphasis on source speaker style transfer, while the HW-TSC S2ST model is built using a cascaded ASR and MT system. It is important to note that the results are not directly comparable due to differences in the ASR models used for evaluation. However, we observe that the models with CPT show performance potentially comparable to the cascaded approach, highlighting the effectiveness of CPT for this task.

6 Conclusion

We explore continual pre-training as an effective strategy to extend codec-based speech LLMs for speech translation-related tasks. By carefully formulating our pre-training data, we adapt a pre-trained textual LLM in two configurations—one with speech-only data and another with a joint speech-text approach. Our extensive experiments on ASR, TTS, S2TT, and S2ST tasks show that continual pre-training can significantly enhance performance. In particular, speech-only continual pre-training yields notable improvements for TTS, while joint speech-text continual pre-training strikes a balance between understanding and generation, ultimately delivering high-quality end-to-end S2ST. These findings underscore the potential of continual pre-training in addressing issues such as catastrophic forgetting and modality mismatch, thereby advancing the development of robust multimodal language models.[11]

[10] Due to the non-convergence, we did not put the results in Table 4.

[11] Some generated audio samples are available at https://hiddenmeprivate.github.io/

7 Limitations

We demonstrate through several experiments that CPT can potentially help balance the generation and understanding abilities in codec-based speech LLMs. However, we acknowledge the following limitations:

- Limited model size: Due to computational and time constraints, we were unable to train a larger-scale model. In comparison to many recent models with 7B or more parameters, the smaller capacity of our model may limit the full exploration of our proposed approach.
- Difficulty in comparison to recent works: Performing fair comparisons with recent models is challenging for several reasons, such as mismatched pre-trained textual LLMs (which we could not utilize due to computational limitations), mismatched datasets, and the difficulty of reproducing results with limited resources. As a result, most reference models cannot be directly compared to our models.
- Use of in-house data: For continual pre-training, we employed some in-house data to reduce bias present in open-source data (e.g., language distribution, task distribution, and demographic distribution). However, due to privacy agreements and licensing issues, this data cannot be openly shared.
- Comprehensiveness of ablation studies: Given our computational budgets, we were unable to conduct full-cycle ablation studies for all aspects of the proposed methodology. Instead, we modified and tested design choices during early-stage training, where performance may not be fully representative of the later-stage training results.

8 Ethical Statement and Potential Risks

The development of codec-based speech language models through CPT has the potential to enhance a wide range of speech-related tasks, including speech-to-text and text-to-speech translations, thereby contributing to advancements in communication technologies. However, the use of these models must be approached with caution due to several ethical considerations:

Speech data often contains sensitive personal information. The use of speech language models necessitates strict adherence to privacy laws and regulations, such as GDPR, to ensure that personally identifiable information is not inadvertently exposed or misused. Robust mechanisms for data anonymization and secure storage must be employed to prevent unauthorized access or exploitation of individuals' speech data.

The ability to generate high-quality speech through TTS or S2ST poses risks related to the generation of deepfake audio or other forms of speech-based manipulation. Misuse of this technology could potentially lead to the spread of misinformation or impersonation, raising concerns over its potential role in fraudulent activities or the dissemination of false information. Measures such as watermarking or other forms of verifiable speech generation could be explored to mitigate these risks.

In developing codec-based speech language models, we commit to adhering to ethical principles that prioritize user privacy, data security, and fairness, while also actively working to mitigate any negative societal impacts that may arise from misuse of this technology.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Philip Anastassiou, Jiawei Chen, Jitong Chen, Yuanzhe Chen, Zhuo Chen, Ziyi Chen, Jian Cong, Lelai Deng, Chuang Ding, Lu Gao, et al. 2024. Seed-TTS: A family of high-quality versatile speech generation models. *arXiv preprint arXiv:2406.02430*.

Rosana Ardila, Megan Branson, Kelly Davis, Michael Kohler, Josh Meyer, Michael Henretty, Reuben Morais, Lindsay Saunders, Francis Tyers, and Gregor Weber. 2020. Common voice: A massively-multilingual speech corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4218–4222, Marseille, France. European Language Resources Association.

Hui Bu, Jiayu Du, Xingyu Na, Bengu Wu, and Hao Zheng. 2017. Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline. In *2017 20th conference of the oriental chapter of the international coordinating committee on speech databases and speech I/O systems and assessment (O-COCOSDA)*, pages 1–5. IEEE.

Kai-Wei Chang, Haibin Wu, Yu-Kai Wang, Yuan-Kuei Wu, Hua Shen, Wei-Cheng Tseng, Iu-thing Kang, Shang-Wen Li, and Hung-yi Lee. 2024a. Speechprompt: Prompting speech language models for

speech processing tasks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Xuankai Chang, Jiatong Shi, Jinchuan Tian, Yuning Wu, Yuxun Tang, Yihan Wu, Shinji Watanabe, Yossi Adi, Xie Chen, and Qin Jin. 2024b. The interspeech 2024 challenge on speech processing using discrete units. In *Interspeech 2024*, pages 2559–2563.

Xuankai Chang, Brian Yan, Kwanghee Choi, Jee-Weon Jung, Yichen Lu, Soumi Maiti, Roshan Sharma, Jiatong Shi, Jinchuan Tian, Shinji Watanabe, et al. 2024c. Exploring speech recognition, translation, and understanding with discrete speech units: A comparative study. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11481–11485. IEEE.

Xuankai Chang, Brian Yan, Yuya Fujita, Takashi Maekaku, and Shinji Watanabe. 2023. Exploration of efficient end-to-end ASR using discretized input from self-supervised learning. In *INTERSPEECH 2023*, pages 1399–1403.

Guoguo Chen, Shuzhou Chai, Guan-Bo Wang, Jiayu Du, Wei-Qiang Zhang, Chao Weng, Dan Su, Daniel Povey, Jan Trmal, Junbo Zhang, Mingjie Jin, Sanjeev Khudanpur, Shinji Watanabe, Shuaijiang Zhao, Wei Zou, Xiangang Li, Xuchen Yao, Yongqing Wang, Zhao You, and Zhiyong Yan. 2021. Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio. In *Interspeech 2021*, pages 3670–3674.

Sanyuan Chen, Shujie Liu, Long Zhou, Yanqing Liu, Xu Tan, Jinyu Li, Sheng Zhao, Yao Qian, and Furu Wei. 2024a. VALL-E 2: Neural codec language models are human parity zero-shot text to speech synthesizers. *arXiv preprint arXiv:2406.05370*.

Zhehuai Chen, He Huang, Andrei Andrusenko, Oleksii Hrinchuk, Krishna C Puvvada, Jason Li, Subhankar Ghosh, Jagadeesh Balam, and Boris Ginsburg. 2024b. SALM: Speech-augmented language model with in-context learning for speech recognition and translation. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 13521–13525. IEEE.

Zhehuai Chen, Yu Zhang, Andrew Rosenberg, Bhuvana Ramabhadran, Pedro J. Moreno, Ankur Bapna, and Heiga Zen. 2022. MAESTRO: Matched speech text representations through modality matching. In *Interspeech 2022*, pages 4093–4097.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, Chang Zhou, and Jingren Zhou. 2024a. Qwen2-audio technical report. *Preprint*, arXiv:2407.10759.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, et al. 2024b. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. 2023. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. *Preprint*, arXiv:2311.07919.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Defossez. 2023. Simple and controllable music generation. In *Advances in Neural Information Processing Systems*, volume 36, pages 47704–47720. Curran Associates, Inc.

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

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

Kunal Dhawan, Nithin Rao Koluguri, Ante Jukić, Ryan Langman, Jagadeesh Balam, and Boris Ginsburg. 2024. Codec-asr: Training performant automatic speech recognition systems with discrete speech representations. In *Interspeech 2024*, pages 2574–2578.

Jiayu Du, Xingyu Na, Xuechen Liu, and Hui Bu. 2018. Aishell-2: Transforming mandarin asr research into industrial scale. *arXiv preprint arXiv:1808.10583*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Yuan Gong, Alexander H Liu, Hongyin Luo, Leonid Karlinsky, and James Glass. 2023. Joint audio and speech understanding. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8. IEEE.

Yuan Gong, Hongyin Luo, Alexander H. Liu, Leonid Karlinsky, and James R. Glass. 2024. Listen, think, and understand. In *The Twelfth International Conference on Learning Representations*.

Yiwei Guo, Zhihan Li, Hankun Wang, Bohan Li, Chongtian Shao, Hanglei Zhang, Chenpeng Du, Xie Chen, Shujie Liu, and Kai Yu. 2025. Recent advances in discrete speech tokens: A review. *arXiv preprint arXiv:2502.06490*.

Chi Han, Mingxuan Wang, Heng Ji, and Lei Li. 2021. Learning shared semantic space for speech-to-text

translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2214–2225, Online. Association for Computational Linguistics.

Haorui He, Zengqiang Shang, Chaoren Wang, Xuyuan Li, Yicheng Gu, Hua Hua, Liwei Liu, Chen Yang, Jiaqi Li, Peiyang Shi, et al. 2024. Emilia: An extensive, multilingual, and diverse speech dataset for large-scale speech generation. *arXiv preprint arXiv:2407.05361*.

François Hernandez, Vincent Nguyen, Sahar Ghannay, Natalia Tomashenko, and Yannick Esteve. 2018. TED-LIUM 3: Twice as much data and corpus repartition for experiments on speaker adaptation. In *Speech and Computer: 20th International Conference, SPECOM 2018*, pages 198–208. Springer.

Shujie Hu, Long Zhou, Shujie Liu, Sanyuan Chen, Hongkun Hao, Jing Pan, Xunying Liu, Jinyu Li, Sunit Sivasankaran, Linquan Liu, et al. 2024. Wavllm: Towards robust and adaptive speech large language model. *arXiv preprint arXiv:2404.00656*.

Chien-yu Huang, Wei-Chih Chen, Shu-wen Yang, Andy T Liu, Chen-An Li, Yu-Xiang Lin, Wei-Cheng Tseng, Anuj Diwan, Yi-Jen Shih, Jiatong Shi, et al. 2024a. Dynamic-SUPERB phase-2: A collaboratively expanding benchmark for measuring the capabilities of spoken language models with 180 tasks. *arXiv preprint arXiv:2411.05361*.

Chien-yu Huang, Ke-Han Lu, Shih-Heng Wang, Chi-Yuan Hsiao, Chun-Yi Kuan, Haibin Wu, Siddhant Arora, Kai-Wei Chang, Jiatong Shi, Yifan Peng, et al. 2024b. Dynamic-SUPERB: Towards a dynamic, collaborative, and comprehensive instruction-tuning benchmark for speech. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12136–12140. IEEE.

Rongjie Huang, Chunlei Zhang, Yongqi Wang, Dongchao Yang, Jinchuan Tian, Zhenhui Ye, Luping Liu, Zehan Wang, Ziyue Jiang, Xuankai Chang, et al. 2024c. Make-a-voice: Revisiting voice large language models as scalable multilingual and multitask learners. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10929–10942.

Wen Chin Huang, Erica Cooper, Yu Tsao, Hsin-Min Wang, Tomoki Toda, and Junichi Yamagishi. 2022. The VoiceMOS challenge 2022. In *Interspeech 2022*, pages 4536–4540.

Zhichao Huang, Rong Ye, Tom Ko, Qianqian Dong, Shanbo Cheng, Mingxuan Wang, and Hang Li. 2023. Speech translation with large language models: An industrial practice. *arXiv preprint arXiv:2312.13585*.

Jee-Weon Jung, Wangyou Zhang, Jiatong Shi, Zakaria Aldeneh, Takuya Higuchi, Alex Gichamba, Barry-John Theobald, Ahmed Hussen Abdelaziz, and Shinji Watanabe. 2024. ESPnet-SPK: full pipeline speaker embedding toolkit with reproducible recipes, self-supervised front-ends, and off-the-shelf models. In *Interspeech 2024*, pages 4278–4282.

Eugene Kharitonov, Ann Lee, Adam Polyak, Yossi Adi, Jade Copet, Kushal Lakhotia, Tu Anh Nguyen, Morgane Riviere, Abdelrahman Mohamed, Emmanuel Dupoux, 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)*, pages 8666–8681.

Jaehyeon Kim, Keon Lee, Seungjun Chung, and Jaewoong Cho. 2024. CLam-TTS: Improving neural codec language model for zero-shot text-to-speech. In *The Twelfth International Conference on Learning Representations*.

Minje Kim and Jan Skoglund. 2024. Neural speech and audio coding. *Preprint*, arXiv:2408.06954.

Vijay Anand Korthikanti, Jared Casper, Sangkug Lym, Lawrence McAfee, Michael Andersch, Mohammad Shoeybi, and Bryan Catanzaro. 2023. Reducing activation recomputation in large transformer models. *Proceedings of Machine Learning and Systems*, 5:341–353.

Chun-Yi Kuan, Chih-Kai Yang, Wei-Ping Huang, Ke-Han Lu, and Hung-yi Lee. 2024. Speech-Copilot: Leveraging large language models for speech processing via task decomposition, modularization, and program generation. *arXiv preprint arXiv:2407.09886*.

Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. 2023. High-fidelity audio compression with improved rvqgan. *ArXiv*, abs/2306.06546.

Ann Lee, Peng-Jen Chen, Changhan Wang, Jiatao Gu, Sravya Popuri, Xutai Ma, Adam Polyak, Yossi Adi, Qing He, Yun Tang, Juan Pino, and Wei-Ning Hsu. 2022. Direct speech-to-speech translation with discrete units. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3327–3339, Dublin, Ireland. Association for Computational Linguistics.

Soumi Maiti, Yifan Peng, Shukjae Choi, Jee-weon Jung, Xuankai Chang, and Shinji Watanabe. 2024. VoxtLM: Unified decoder-only models for consolidating speech recognition, synthesis and speech, text continuation tasks. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 13326–13330. IEEE.

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. IEEE.

Yifan Peng, Jinchuan Tian, Brian Yan, Dan Berrebbi, Xuankai Chang, Xinjian Li, Jiatong Shi, Siddhant

Arora, William Chen, Roshan Sharma, et al. 2023. Reproducing Whisper-style training using an open-source toolkit and publicly available data. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–8. IEEE.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Vineel Pratap, Qiantong Xu, Anuroop Sriram, Gabriel Synnaeve, and Ronan Collobert. 2020. MLS: A large-scale multilingual dataset for speech research. In *Interspeech 2020*, pages 2757–2761.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.

J Rottland, Ch Neukirchen, and D Willett. 1997. The WSJ speech database.

Paul K Rubenstein, Chulayuth Asawaroengchai, Duc Dung Nguyen, Ankur Bapna, Zalán Borsos, Félix de Chaumont Quitry, Peter Chen, Dalia El Badawy, Wei Han, Eugene Kharitonov, et al. 2023. AudioPaLM: A large language model that can speak and listen. *arXiv preprint arXiv:2306.12925*.

Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. 2022. UTMOS: Utokyo-sarulab system for voicemos challenge 2022. In *Interspeech 2022*, pages 4521–4525.

Jiatong Shi, Yun Tang, Ann Lee, Hirofumi Inaguma, Changhan Wang, Juan Pino, and Shinji Watanabe. 2023. Enhancing speech-to-speech translation with multiple TTS targets. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Jiatong Shi, Jinchuan Tian, Yihan Wu, Jee weon Jung, Jia Qi Yip, Yoshiki Masuyama, William Chen, Yuning Wu, Yuxun Tang, Massa Baali, Dareen Alharhi, Dong Zhang, Ruifan Deng, Tejes Srivastava, Haibin Wu, Alexander H. Liu, Bhiksha Raj, Qin Jin, Ruihua Song, and Shinji Watanabe. 2024. ESPnet-Codec: Comprehensive training and evaluation of neural codecs for audio, music, and speech. *Preprint*, arXiv:2409.15897.

Yao Shi, Hui Bu, Xin Xu, Shaoji Zhang, and Ming Li. 2021. Aishell-3: A multi-speaker mandarin tts corpus. In *Interspeech 2021*, pages 2756–2760.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2020. Ernie 2.0: A continual pre-training framework for language understanding. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8968–8975.

Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun MA, and Chao Zhang. 2024a. SALMONN: Towards generic hearing abilities for large language models. In *The Twelfth International Conference on Learning Representations*.

Yuxun Tang, Yuning Wu, Jiatong Shi, and Qin Jin. 2024b. SingOMD: Singing oriented multi-resolution discrete representation construction from speech models. In *Interspeech 2024*, pages 2564–2568.

Jinchuan Tian, Chunlei Zhang, Jiatong Shi, Hao Zhang, Jianwei Yu, Shinji Watanabe, and Dong Yu. 2024. Preference alignment improves language model-based TTS. *arXiv preprint arXiv:2409.12403*.

Changhan Wang, Anne Wu, Jiatao Gu, and Juan Pino. 2021. CoVoST 2 and massively multilingual speech translation. In *Interspeech 2021*, pages 2247–2251.

Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, et al. 2023. Neural codec language models are zero-shot text to speech synthesizers. *arXiv preprint arXiv:2301.02111*.

Wei Wang, Shuo Ren, Yao Qian, Shujie Liu, Yu Shi, Yanmin Qian, and Michael Zeng. 2022. Optimizing alignment of speech and language latent spaces for end-to-end speech recognition and understanding. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7802–7806. IEEE.

Xiaofei Wang, Manthan Thakker, Zhuo Chen, Naoyuki Kanda, Sefik Emre Eskimez, Sanyuan Chen, Min Tang, Shujie Liu, Jinyu Li, and Takuya Yoshioka. 2024. Speechx: Neural codec language model as a versatile speech transformer. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Haibin Wu, Ho-Lam Chung, Yi-Cheng Lin, Yuan-Kuei Wu, Xuanjun Chen, Yu-Chi Pai, Hsiu-Hsuan Wang, Kai-Wei Chang, Alexander H. Liu, and Hung yi Lee. 2024a. Codec-SUPERB: An in-depth analysis of sound codec models. *Preprint*, arXiv:2402.13071.

Tongtong Wu, Linhao Luo, Yuan-Fang Li, Shirui Pan, Thuy-Trang Vu, and Gholamreza Haffari. 2024b. Continual learning for large language models: A survey. *Preprint*, arXiv:2402.01364.

Yihan Wu, Soumi Maiti, Yifan Peng, Wangyou Zhang, Chenda Li, Yuyue Wang, Xihua Wang, Shinji Watanabe, and Ruihua Song. 2024c. SpeechComposer: Unifying multiple speech tasks with prompt composition. *Preprint*, arXiv:2401.18045.

Zhanglin Wu, Jiaxin Guo, Daimeng Wei, Zhiqiang Rao, Zongyao Li, Hengchao Shang, Yuanchang Luo, Shaojun Li, and Hao Yang. 2024d. Improving the quality of IWLST 2024 cascade offline speech translation and speech-to-speech translation via translation hypothesis ensembling with nmt models and large language models. In *Proceedings of the 21st International Conference on Spoken Language Translation (IWSLT 2024)*, pages 46–52.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024a. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Dongchao Yang, Jinchuan Tian, Xu Tan, Rongjie Huang, Songxiang Liu, Haohan Guo, Xuankai Chang, Jiatong Shi, Jiang Bian, Zhou Zhao, et al. 2024b. UniAudio: Towards universal audio generation with large language models. In *Forty-first International Conference on Machine Learning*.

Rong Ye, Chengqi Zhao, Tom Ko, Chutong Meng, Tao Wang, Mingxuan Wang, and Jun Cao. 2023. Gigast: A 10,000-hour pseudo speech translation corpus. In *INTERSPEECH 2023*, pages 2168–2172.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. 2021. 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. In *Interspeech 2019*, pages 1526–1530.

Yuexiang Zhai, Shengbang Tong, Xiao Li, Mu Cai, Qing Qu, Yong Jae Lee, and Yi Ma. 2023. Investigating the catastrophic forgetting in multimodal large language model fine-tuning. In *Conference on Parsimony and Learning (Proceedings Track)*.

Jun Zhan, Junqi Dai, Jiasheng Ye, Yunhua Zhou, Dong Zhang, Zhigeng Liu, Xin Zhang, Ruibin Yuan, Ge Zhang, Linyang Li, et al. 2024. AnyGPT: Unified multimodal LLM with discrete sequence modeling. *arXiv preprint arXiv:2402.12226*.

Binbin Zhang, Hang Lv, Pengcheng Guo, Qijie Shao, Chao Yang, Lei Xie, Xin Xu, Hui Bu, Xiaoyu Chen, Chenchen Zeng, et al. 2022. Wenetspeech: A 10000+ hours multi-domain mandarin corpus for speech recognition. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6182–6186. IEEE.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2023a. SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15757–15773.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2023b. SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15757–15773, Singapore. Association for Computational Linguistics.

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*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Xinfa Zhu, Yuanjun Lv, Yi Lei, Tao Li, Wendi He, Hongbin Zhou, Heng Lu, and Lei Xie. 2023. VecSok speech: Speech vectorization and tokenization for neural speech generation. *arXiv preprint arXiv:2310.07246*.

A Speech Tokenizer

The tokenizer is optimized with 8-stream ($L = 8$) residual vector quantization (RVQ) layers, each containing 1,024 tokens per codebook ($|\mathcal{B}_j| = 1,024$, $(j = 1, ..., L)$). The framerate is set to 50Hz. Most hyperparameters related to the model architecture are aligned with those in the original SoundStream paper (Zeghidour et al., 2021), while the discriminator setups follow the DAC framework (Kumar et al., 2023).

We use a segment size of 24,000 samples (1.5 seconds) with a batch size of 6. The loss terms are consistent with those in the DAC paper. Both the generator and discriminator are optimized using the AdamW optimizer with a learning rate of 0.0002. We apply an exponential learning rate scheduler with a decay rate of 0.999.

Due to the nature of RVQ, the initial streams in the codec typically carry the bulk of the signal information. To further enhance learning in these initial streams, we select the target bandwidth from 0.5, 1, 1.5, 2.0, 4.0, sampled uniformly. Empirically, this setup achieves better reconstruction quality using only three codec levels ($L' = 3$), reducing the modeling complexity in the codec-based speech LLM. We also introduce noise and reverberation into 20% of the training data to improve the model's understanding capabilities. The signal-to-noise ratio (SNR) for these samples is randomly

chosen from a range of 6.0 to 20.0 dB. The total training steps are 1.6M.

B Pre-training Dataset

B.1 Pre-training Data Details

The list of pre-training data is provided in Table 5.

B.2 Pre-training Data Simulation

Among recent speech LLMs, we identify two approaches to support multi-task training. One approach uses natural language prompts, enabling generalization across multiple tasks (Tang et al., 2024a; Gong et al., 2024; Chu et al., 2024a; Hu et al., 2024). These prompts can also serve as a bridge between the newly introduced speech modality and the original text, particularly when a pre-trained text LLM is involved.

The second approach uses a task template with either explicit task tokens or modality orders (Chu et al., 2023; Yang et al., 2024b; Wang et al., 2023; Chen et al., 2024b). While this method allows for more stable modeling across different tasks, it is less flexible. Most generation-oriented speech LLMs use this approach.

In our work, to best leverage the reasoning capabilities of the pre-trained text LLM, we opt for the natural language prompt method to connect different modalities.

To generate prompts with enough variations, we use OpenAI APIs for both ChatGPT powered by either GPT3.5 or GPT-4 (Achiam et al., 2023). The prompts are generated in both English and Mandarin. After the initial generation of 50 prompts for both English and Mandarin, we conduct manual filtering to remove unreasonable prompts, resulting in 25 English prompts and 25 Mandarin prompts per task. For the prompts used during pre-training, we limit the prompts to have a declarative format during initial generation, while we specify multiple formats for the prompts used during fine-tuning, including declaratives, interrogatives, and imperatives.

We define the pre-training data simulation policy as follows:

- For ASR, S2TT, TTS, and T2ST, the randomly selected natural language prompt can be in either language, regardless of the language used in the corresponding speech-text pair.
- For each sample in the TTS task, a portion of the target codec tokens is randomly selected, with a duration ranging from $[\min(T_C/4, 2\text{seconds}), \min(T_C/2, 4\text{seconds})]$. Additionally, we use ChatGPT to generate natural language prompts that specify the task of synthesizing speech in the speaking style of the target speech. These prompts are written in an imperative format, matching the language of the task-specific prompt, and are concatenated with the input text and acoustic conditioning in the following format: '*<Text><Speaking Style Prompt><Acoustic Conditioning><Task Prompt>*'.

C Model Hyper-parameters

Pre-training experiments are conducted with tensor parallelism set to 8, which enables the use of larger batch sizes during pre-training (Korthikanti et al., 2023). The gradient clip is set to 1.0. The normalization epsilon is set to 1e-5. The global batch size is set to 640 with a sample sequence length of 4,096. The number of training steps is set to 40k. The model is trained on `BFloat16`. We use the distributed Adamw optimizer with a peak learning rate of 1e-5 and a minimum learning rate of 1e-6. The expanded vocabulary size is 155,012, considering padding tokens and 293 additional tokens for tensor shape adjustment to achieve tensor parallel training. The model has a parameter size of 943.5M.

Table 5: Continual pre-training dataset.

Dataset	Language	Data Type	Data Size (Hour)
Aishell{1-3} (Bu et al., 2017; Du et al., 2018; Shi et al., 2021)	ZH	Read	1,200
Wenetspeech (Zhang et al., 2022)	ZH	Various	10,000
Gigaspeech (Chen et al., 2021)	EN	Various	10,000
Librispeech (Panayotov et al., 2015)	EN	Read	1,000
MLS (Pratap et al., 2020)	EN	Read	44,000
TEDLIUM3 (Hernandez et al., 2018)	EN	Lecture	400
WSJ (Rottland et al., 1997)	EN	Read	140
Commonvoice (Ardila et al., 2020)	EN & ZH	Read	2,600
In-house	ZH	Various	70,000
Crawled Youtube subtitles	EN & ZH	-	-
Wikipedia	EN	-	-
The pile book corpus (Gao et al., 2020)	EN	-	-
In-house Mandarin data	ZH	-	-
In-house translation data	EN & ZH	-	-