Lightweight Language Model for Speech Synthesis: Attempts and Analysis

Zhuojun Wu[1,2]*,Dong Liu*[1,2]*, Ming Li*[1,2]

[1]School of Computer Science, Wuhan University, Wuhan, China
[2]Suzhou Municipal Key Laboratory of Multimodal Intelligent Systems,
Data Science Research Center, Duke Kunshan University, Kunshan, China

zhuojun.wu@dukekunshan.edu.cn, dong.liu@dukekunshan.edu.cn,
ming.li369@dukekunshan.edu.cn

Abstract

Large-scale autoregressive text-to-speech (TTS) models can generate speech that is nearly indistinguishable from human speech. However, training large language models (LLMs) is challenging due to memory and computational constraints. This paper describes our TTS method for the 2024 Conversational Voice Clone Challenge (CoVoC). Our approach modifies the LauraGPT model to synthesize mixed Chinese and English text by expanding the Chinese pinyin vocabulary and reducing the number of layers in the decoder-only Transformer architecture. Despite using minimal training data, the performance gap between our method and other constrained systems is relatively small in both subjective and some objective evaluations. This paper discusses our attempt to train lightweight LLMs for zero-shot TTS and analyzes the factors contributing to low performance. Our audio samples can be accessed online[1].

Index Terms: text-to-speech, speech recognition, human-computer interaction, computational paralinguistics

1. Introduction

Text-to-speech (TTS) aims to produce natural and human-like speech from provided text. Recent advancements in deep neural network-based TTS have significantly improved the field. However, developing a speech synthesis system that achieves high expressiveness, naturalness, and speaker similarity remains challenging.

Numerous methods have been proposed, significantly improving TTS applications and transforming human-machine interactions. Previous works introduced a two-stage TTS system: the first stage[1, 2, 3, 4, 5, 6, 7] generates intermediate speech representations, such as mel-spectrograms or linguistic features from the input text, and the second stage[8, 9, 10, 11, 12, 13] produces raw waveforms based on these representations. However, this two-stage approach can lead to error propagation and relies on human-defined features, limiting the system's performance. To address these issues, single-stage models[14, 15] that directly generate waveforms from the input text have been proposed. Although these models have shown great success, they often struggle with speaker similarity and have difficulties in achieving fully human-like speech.

Recent advancements in neural speech synthesis have significantly improved the quality and naturalness of synthesized speech. The increase in available data[16] and the success of large language models (LLMs) in natural language processing (NLP) have inspired new approaches. A notable development in this field is zero-shot TTS, where systems generate speech for unseen speakers without additional training. TTS systems based on LLMs[17, 18] have shown remarkable abilities in zero-shot generalization. Wang et al.[17] introduced VALL-E, which uses Encodec's encoder[19] to process waveform features and a language model to predict output tokens, followed by Encodec's decoder to generate waveforms. This method demonstrates high speaker similarity using just one utterance from an unseen speaker. The increased scale of model size and enriched training data have advanced generative modeling, particularly in text and image generation[20]. Recently, speech synthesis also starts to demand larger models and datasets. Early results from large-scale TTS models have demonstrated capabilities such as zero-shot speaker adaptation[17].

However, using large language models like LLaMA[21] is often constrained by limited computational resources and financial costs[22], making them unsuitable for personal devices with lower compute power and GPU memory. To address these limitations, we propose a zero-shot TTS system based on a lightweight LLM. First, we use a codec encoder to obtain discretized tokens. Our lightweight LLM then generates output tokens based on the text, and finally, a codec decoder generates waveforms from these tokens. To mitigate errors in speech generated from lengthy texts, we introduce a chunking method. Despite using minimal training data and computational resources, our system achieves results comparable to other systems in both subjective and some objective evaluations. This approach suggests that LLM-based TTS systems can be developed even with limited training data and computational power.

2. Proposed Method

Our proposed speech synthesis system is built on the LauraGPT model[23] and is designed to synthesize high-quality audio from textual input, supporting both Chinese and English languages. Initially, the pre-trained LauraGPT model was only available for English text-to-speech synthesis. We expanded the phoneme vocabulary to encompass Chinese pinyin phonemes, enabling the system to perform mixed-language text-to-speech synthesis.

2.1. Audio Codec

We utilized an open-source, pre-trained codec model named audio-codec-encodeczh-en-general-16k-nq32ds640-pytorch from the Funcodec toolkit[24]. This codec employs 32 codebooks, downsamples the original audio by a factor of 640, and supports both Chinese and English speech at a sampling rate of 16KHz. The model is a variational autoencoder (VAE)[25] that uses residual vector quantization (RVQ)[26] to generate several parallel sequences of discrete latent representations. Trained on a large-scale dataset[24], it is robust across various scenarios.

Corresponding Author: Ming Li

[1]https://axunyi.github.io/lwllmtts/

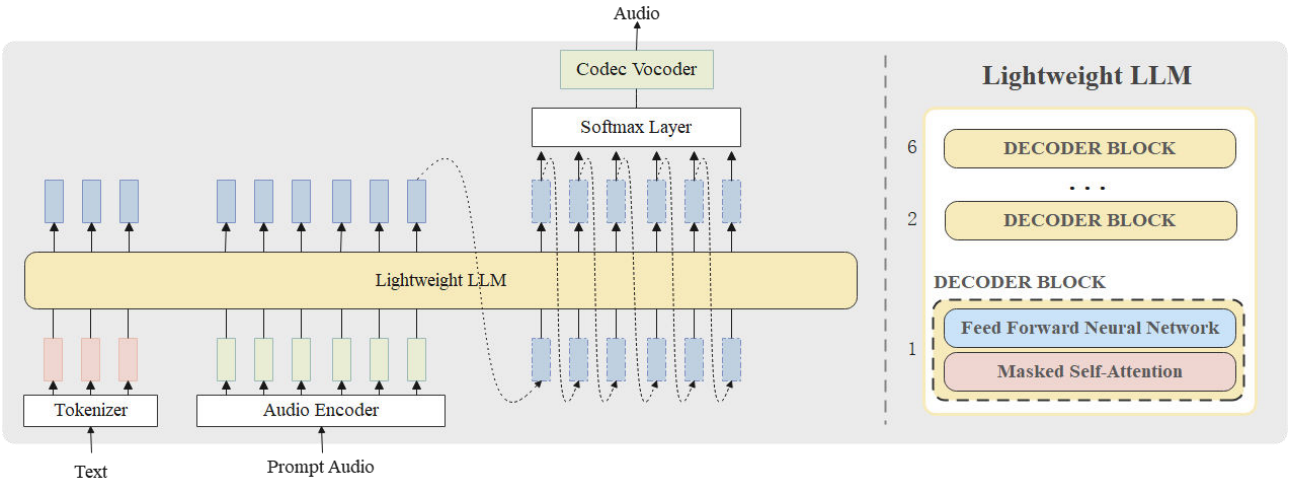

Figure 1: *An overview of our proposed system*

Compared to Encodec[19] and SoundStream[26], this model uses magnitude spectrum loss to enhance middle and high-frequency signals. It also integrates structured dropout to smooth the code space and accommodate various bandwidths within a single model. Instead of random initialization, it initializes codes using k-means clustering for a more systematic approach. Additionally, the model maintains codebooks with an exponential moving average and a dead-code-elimination mechanism, ensuring a high utilization factor for the codebooks. These innovations collectively contribute to the model's superior performance in speech processing.

2.2. Lightweight LLM

For LLMs, considering that constraints often encounter in devices with limited GPU memory, we try to optimize our model for such environments. Specifically, we develop a compact model by downsizing the decoder-only Transformer components. This was achieved by reducing the number of layers from 12 to 6, resulting in a substantial decrease in the model's parameter count from 40 million to 20 million. This reduction in size ensures the model remains effective and efficient, even under resource-constrained conditions. The model structure is shown on the right side of Figure1.

2.3. Model Structure

The structure of our model is shown on the left side of Figure1. It comprises three components: a lightweight LLM backbone, an audio encoder, and a codec vocoder. For audio inputs, we feed them into the audio encoder, while the audio outputs are discretized into tokens using the audio tokenizer. Text data, both inputs, and outputs are processed by the text tokenizer, which is an open-source tokenizer from Qwen[27]. The LLM then predicts the next audio tokens. Finally, we use a one-step codec vocoder to convert audio tokens into raw waveforms.

2.4. Chunking

The aforementioned method is the one we submitted to the CoVoC challenge. Due to time constraints, there were some issues with the results we submitted. After the submission, we dedicated efforts to resolve some of these problems, particularly those related to long text audio synthesis. In our initial submission, all text was input directly, regardless of length. This approach led to poor quality of long audio due to the degradation problem of the LLM when handling extended sequences[28]. After the submission deadline, we have made progress in addressing this issue.

To handle lengthy input texts more effectively, we adopt a chunking mechanism. The primary objective is to enhance the stability and effectiveness of the audio generation process, particularly for texts that exceed the typical length parameters.

Our approach involves splitting the input text into smaller, but more manageable chunks. This method helps reduce the impact of LLM degradation caused by synthesizing long audio sequences. By breaking down the text, we generate audio sequentially, sentence by sentence. This allows our system to focus on the precise phonetic and intonational nuances of each segment, ensuring higher accuracy and natural fluency in the audio output.

The sentence-by-sentence approach offers significant advantages in terms of error management and control. Deviding the generation process means that any potential discrepancies or inaccuracies can be more readily identified and addressed at the sentence level, rather than having to review and revise an entire block of generated audio. This granularity in error detection and correction enhances the overall quality of the synthesized speech.

2.5. Inference

During the inference stage, audio inputs are first fed into the audio encoder to generate a sequence of tokens. Simultaneously, any accompanying text data is chunked, then processed by the text tokenizer. The LLM then leverages these tokens to predict the subsequent audio tokens in the sequence. Once the LLM has completed its predictions, the codec vocoder converts the predicted tokens into a continuous waveform that can be played as natural-sounding speech.

3. Experiments

3.1. Data

The CoVoC challenge encompasses a range of diverse training datasets, including the 10K-hour WenetSpeech4TTS[29] dataset, 180 hours of Mandarin spontaneous conversational speech data, and 100 hours of high-quality spoken conversations. All the audio data is in mono-channel WAV format with a sample rate of 16KHz.

Specifically, WenetSpeech4TTS is a large-scale, multi-domain Mandarin dataset with accurate transcriptions. The MAGIC DATA Mandarin Chinese Conversational Speech Corpus contains 180 hours of mobile-recorded speech data, featuring coherent conversational data labeled with non-verbal information such as laughter and coughs, as well as speech or mixed speech that is difficult to distinguish. HQ-Conversations includes 100 hours of natural and expressive speech from 200 speakers. For the Constrained Track, we are only allowed to train with the above datasets.

For our system, we need a training dataset that is both segmented and speaker-specific. Therefore, we selected a subset of the WenetSpeech4TTS dataset for the initial pretraining phase of our model. The WenetSpeech4TTS dataset is a comprehensive resource that includes a wide variety of speech data. Specifically, we only use the Premium subset, which comprises approximately 945 hours of speech data. This subset contains speech data with DNSMOS[30] greater than 4.0, and we hypothesized that using higher-quality speech data can effectively reduce the amount of required pre-training data. The quality and diversity of the data within this subset are conducive to developing a model that can synthesize speech that is both natural and clear.

Following the pretraining phase, we fine-tuned our model using the 100-hour HQ-Conversations dataset. The goal of this fine-tuning process was to further enhance the naturalness and expressiveness of the synthesized speech. By training on this dataset, our model could better capture the subtleties of human speech, such as intonation, rhythm, and emotional nuances.

Our approach to training involves a combination of pre-training on a large, diverse dataset followed by fine-tuning on a smaller, high-quality dataset. This method is designed to optimize both the efficiency and effectiveness of our speech synthesis system. It allows us to balance the need for a comprehensive training experience with the practical constraints of computational resources and training time.

3.2. Training Setting

In our research, we used the dataset provided by the challenge for system development. Specifically, we utilized a pre-trained codec model to extract discrete tokens and kept the model parameters frozen without any further training. For training our decoder-only Transformer backend, we initially pre-trained the model using the Premium subset of WenetSpeech4TTS, which consists of 945 hours of speech data.

Subsequently, we fine-tuned our model using the 100-hour HQ-Conversations dataset. During the pre-training phase, we set the learning rate to 0.001, warmup steps to 10,000, and used a batch size of 160, completing a total of 50 epochs. In the fine-tuning phase, we adjusted the initial learning rate to 0.0001 and reduced warmup steps to 100, training for 360 epochs. Other experimental settings remained the same as the default. All training was conducted on an 8 NVIDIA A6000 GPU server.

3.3. Results of the Submitted System

The test set we used is a standardized dataset provided by the CoVoC organizers. It contains prompt audio from 20 speakers and 430 well-designed texts. These texts include complex synthetic scenarios such as reflective pauses in spoken contexts, superimposed words, long texts, colloquialized phrases, and mixed Chinese and English. These varied and intricate contexts are designed to thoroughly evaluate the TTS system's performance.

The evaluation process consists of both subjective and objective assessments. The subjective evaluation uses the mean opinion score (MOS) to assess four criteria: speech quality (SQ), speech naturalness (SN), speaker similarity (SS), and speech spontaneous style (SSS). The objective evaluation measures include the character error rate (CER) and speaker embedding cosine similarity (SECS). Our subjective and objective evaluation scores for the submitted systems are shown in Table 1.

Table 1: *The evaluation score of our submitted system under the constrained task and the official baseline in the unconstrained task*

Criteria	Before Deadline (Constrained)	Official Baseline (Unconstrained)	After Deadline (Constrained)
SN	3.41 ± 0.14	4.07 ± 0.1	
SQ	3.35 ± 0.19	4.16 ± 0.13	
SS	2.88 ± 0.14	3.62 ± 0.1	
SSS	2.95 ± 0.13	3.78 ± 0.1	
CER	34.28%	6.86%	22.22%
SIM	0.76007	0.84866	0.76951

3.4. Post-submission Analysis

3.4.1. Overview

After obtaining the scores for each criterion of our submitted system, we conducted a detailed analysis of the generated test audio. This analysis aimed to identify and resolve the issues that arose during the audio generation phase. We then improved our system as in Section 2.4.

In the absence of subjective evaluations by the organizers, our post-submission analysis is limited to objective measures. While this limitation poses some challenges, it also allows us to provide a targeted and quantitative assessment of system performance.

3.4.2. Prompt Speaker Perspective

After analyzing the objective scores of the audio we submitted, we find that our system performs better on SIM metrics when dealing with longer audio that is clearly articulated. This enhanced performance can be attributed to the wealth of vocal characteristics present in extended audio clips, allowing the system to more accurately capture and replicate the distinctive features of the speaker's voice.

Conversely, the system's performance declines when faced with shorter, more vaguely articulated, or less audible audio. The brevity of the audio may offer insufficient characteristic information, and vague or indistinct pronunciation can lead to the loss or distortion of critical speech features, resulting in decreased performance.

In terms of CER metrics, our system performs particularly poorly when the prompt audio is long or the text contains a large

number of spoken words, especially in the case of mixed Chinese and English input. Long audio and complex linguistic input may increase processing complexity. Mixed-language input increases the uncertainty of the language model because the system has to process the phonetic features and grammatical structures of both languages simultaneously. This can lead to poor performance in tokenization, particularly when the training data is insufficient to cover these linguistic features.

3.4.3. Utterance Perspective

We conducted an in-depth analysis of the corpus of test texts, meticulously designed to simulate the complexities of natural spoken language. These characteristics can lead to undesirable performance of our system.

The length of these texts varies from concise to extensive narratives. A key feature of these texts is the extensive use of colloquial expressions, particularly filler words. Such linguistic elements are ubiquitous in spoken interactions and serve various functions, such as providing pauses in thought, emphasizing certain points, or simply filling in gaps in the flow of conversation. For TTS systems, excessively long input text can lead to degradation of LLM performance[31] and poorer quality of the generated audio. In contrast, the natural generation of spoken expressions relies on the breadth of the training data[32].

Additionally, the texts exhibited a notable prevalence of repetitive words or phrases. This phenomenon is common in spoken language, where repetition can serve to underscore a message or provide a momentary buffer for the speaker to gather thoughts. The presence of such repetitions adds authenticity to the texts, making them more representative of everyday speech patterns. However, for the audio synthesized by our TTS system, it sometimes produces ambiguous patches of repeated words and does not generate the overlapping sound effect found in everyday conversations.

A particularly noteworthy aspect of the text is the mixture of Chinese and English. The inclusion of these two languages not only enriches the text but also poses challenges to language processing and comprehension. Our system can synthesize mixed Chinese and English input due to the expansion of the Chinese and English phoneme vocabulary.

The emotional tone of the text is reflected in various expressions, including skepticism, surprise, and other emotional states. This rich interplay of emotions showcases the text's ability to convey the nuanced and dynamic interplay of human emotions in communication. Since the audio inputs to our system are all calm prompt audio, the synthesized voices are unable to make emotional adjustments in response to the emotional changes in the text.

Faced with synthetic texts featuring intricate spoken language and nuanced emotional expressions, our system performs less well due to being trained on a smaller scale of data. We plan to increase the amount of training data to address these problems.

3.4.4. Results of the chunked system

We found that the selection of chunk size is a critical factor that can significantly influence the outcome of TTS tasks. We conducted experiments to determine the optimal chunk sizes, evaluating their impact on the performance of our language model and the quality of the generated audio. Various chunk sizes were tested to find a balance that minimizes degradation while maintaining the fluidity of the synthesized speech. The results of these experiments are detailed in Table 2 and Table 3.

Table 2: *SIM score of our system at different chunk sizes*

Chunk Size	**Score**
100	0.77205
150	0.76951
200	0.76554

Table 3: *CER score of our system at different chunk sizes*

Chunk Size	**Score**
100	25.88
150	22.22
200	22.65

Combining objective metrics, we found that the best results were obtained when using a chunk size of 150 characters. Further analysis revealed that a too large chunk size reproduces the problem of long text input, leading to performance degradation and making the generation process less effective. Conversely, too small chunk sizes lead to fragmentation of linguistic elements, often segmenting individual words or phrases prematurely. This interference with natural language fluency prevents the model from generating smooth and contextually accurate audio, resulting in poorer outcomes.

After addressing the problem of long text generation, our system achieved an improvement in CER scores.

However, the sentence-by-sentence generation method does have its drawbacks, sometimes resulting in incoherent audio. Despite this, it also presents an opportunity to incorporate natural pauses and intonation more flexibly for inter-sentence affective shifts in future iterations of the system. This potential for dynamic emotional expression is an area we plan to explore further in our future works.

Also, we plan to optimize the system further by scaling up the training data and including emotional speech data. This will help enhance the system's performance and its ability to generate more natural and expressive audio outputs.

4. Conclusions

This paper describes the TTS system with which we participated in the 2024 CoVoC challenge. The system leverages a lightweight LLM to synthesize mixed Chinese and English speech. A chunking method is used to generate audio from lengthy texts. Despite the constraints of minimal training data and computational resources, the proposed method demonstrates competitive performance in both subjective and objective evaluations.

5. Acknowledgements

This research is funded by the Kunshan Municipal Government Research Funding under the project 'Deep Learning based Singing Voice Synthesis for Kun Opera'.

6. References

[1] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," in *2018 IEEE international conference on acoustics,*

speech and signal processing (ICASSP). IEEE, 2018, pp. 4779–4783.

[2] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech: Fast, robust and controllable text to speech," *Advances in neural information processing systems*, vol. 32, 2019.

[3] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu, "Neural speech synthesis with transformer network," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 6706–6713.

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

[5] R. Valle, K. Shih, R. Prenger, and B. Catanzaro, "Flowtron: an autoregressive flow-based generative network for text-to-speech synthesis," *arXiv preprint arXiv:2005.05957*, 2020.

[6] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. Kudinov, "Grad-tts: A diffusion probabilistic model for text-to-speech," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8599–8608.

[7] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech 2: Fast and high-quality end-to-end text to speech," *arXiv preprint arXiv:2006.04558*, 2020.

[8] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[9] K. Kumar, R. Kumar, T. De Boissiere, L. Gestin, W. Z. Teoh, J. Sotelo, A. De Brebisson, Y. Bengio, and A. C. Courville, "Melgan: Generative adversarial networks for conditional waveform synthesis," *Advances in neural information processing systems*, vol. 32, 2019.

[10] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 3617–3621.

[11] M. Bińkowski, J. Donahue, S. Dieleman, A. Clark, E. Elsen, N. Casagrande, L. C. Cobo, and K. Simonyan, "High fidelity speech synthesis with adversarial networks," *arXiv preprint arXiv:1909.11646*, 2019.

[12] R. Yamamoto, E. Song, and J.-M. Kim, "Parallel wavegan: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6199–6203.

[13] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in neural information processing systems*, vol. 33, pp. 17 022–17 033, 2020.

[14] J. Kim, J. Kong, and J. Son, "Conditional variational autoencoder with adversarial learning for end-to-end text-to-speech," in *International Conference on Machine Learning*. PMLR, 2021, pp. 5530–5540.

[15] J. Kong, J. Park, B. Kim, J. Kim, D. Kong, and S. Kim, "Vits2: Improving quality and efficiency of single-stage text-to-speech with adversarial learning and architecture design," *arXiv preprint arXiv:2307.16430*, 2023.

[16] L.-W. Chen, S. Watanabe, and A. Rudnicky, "A vector quantized approach for text to speech synthesis on real-world spontaneous speech," in *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023. [Online]. Available: https://doi.org/10.1609/aaai.v37i11.26488

[17] C. Wang, S. Chen, Y. Wu, Z. Zhang, L. Zhou, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li *et al.*, "Neural codec language models are zero-shot text to speech synthesizers," *arXiv preprint arXiv:2301.02111*, 2023.

[18] Z. Jiang, J. Liu, Y. Ren, J. He, Z. Ye, S. Ji, Q. Yang, C. Zhang, P. Wei, C. Wang, X. Yin, Z. MA, and Z. Zhao, "Mega-TTS 2: Boosting prompting mechanisms for zero-shot speech synthesis," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: https://openreview.net/forum?id=mvMI3N4AvD

[19] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High fidelity neural audio compression," *arXiv preprint arXiv:2210.13438*, 2022.

[20] B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal *et al.*, "Language models are few-shot learners," *arXiv preprint arXiv:2005.14165*, vol. 1, 2020.

[21] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[22] C.-Y. Hsieh, C.-L. Li, C.-K. Yeh, H. Nakhost, Y. Fujii, A. Ratner, R. Krishna, C.-Y. Lee, and T. Pfister, "Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes," *arXiv preprint arXiv:2305.02301*, 2023.

[23] Q. Chen, Y. Chu, Z. Gao, Z. Li, K. Hu, X. Zhou, J. Xu, Z. Ma, W. Wang, S. Zheng *et al.*, "Lauragpt: Listen, attend, understand, and regenerate audio with gpt," *arXiv preprint arXiv:2310.04673*, 2023.

[24] Z. Du, S. Zhang, K. Hu, and S. Zheng, "Funcodec: A fundamental, reproducible and integrable open-source toolkit for neural speech codec," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2024, pp. 591–595.

[25] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[26] N. Zeghidour, A. Luebs, A. Omran, J. Skoglund, and M. Tagliasacchi, "Soundstream: An end-to-end neural audio codec," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 495–507, 2021.

[27] J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, Y. Han, F. Huang, B. Hui, L. Ji, M. Li, J. Lin, R. Lin, D. Liu, G. Liu, C. Lu, K. Lu, J. Ma, R. Men, X. Ren, X. Ren, C. Tan, S. Tan, J. Tu, P. Wang, S. Wang, W. Wang, S. Wu, B. Xu, J. Xu, A. Yang, H. Yang, J. Yang, S. Yang, Y. Yao, B. Yu, H. Yuan, Z. Yuan, J. Zhang, X. Zhang, Y. Zhang, Z. Zhang, C. Zhou, J. Zhou, X. Zhou, and T. Zhu, "Qwen technical report," 2023. [Online]. Available: https://arxiv.org/abs/2309.16609

[28] H. Qin, X. Ma, X. Zheng, X. Li, Y. Zhang, S. Liu, J. Luo, X. Liu, and M. Magno, "Accurate lora-finetuning quantization of llms via information retention," *arXiv preprint arXiv:2402.05445*, 2024.

[29] L. Ma, D. Guo, K. Song, Y. Jiang, S. Wang, L. Xue, W. Xu, H. Zhao, B. Zhang, and L. Xie, "Wenetspeech4tts: A 12,800-hour mandarin tts corpus for large speech generation model benchmark," *arXiv preprint arXiv:2406.05763*, 2024.

[30] C. K. Reddy, V. Gopal, and R. Cutler, "Dnsmos: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6493–6497.

[31] G. Xiao, J. Lin, M. Seznec, H. Wu, J. Demouth, and S. Han, "Smoothquant: Accurate and efficient post-training quantization for large language models," in *International Conference on Machine Learning*. PMLR, 2023, pp. 38 087–38 099.

[32] M. Łajszczak, G. Cámbara, Y. Li, F. Beyhan, A. van Korlaar, F. Yang, A. Joly, Á. Martín-Cortinas, A. Abbas, A. Michalski *et al.*, "Base tts: Lessons from building a billion-parameter text-to-speech model on 100k hours of data," *arXiv preprint arXiv:2402.08093*, 2024.