Interspeech 2024
1-5 September 2024, Kos, Greece

Phonetic Enhanced Language Modeling for Text-to-Speech Synthesis

Kun Zhou, Shengkui Zhao, Yukun Ma, Chong Zhang, Hao Wang, Dianwen Ng, Chongjia Ni, Trung Hieu Nguyen, Jia Qi Yip, Bin Ma

Alibaba Group, Singapore
kun.z@alibaba-inc.com

Abstract

Recent language model-based text-to-speech (TTS) frameworks demonstrate scalability and in-context learning capabilities. However, they suffer from robustness issues due to the accumulation of errors in speech unit predictions during autoregressive language modeling. In this paper, we propose a phonetic enhanced language modeling method to improve the performance of TTS models. We leverage self-supervised representations that are phonetically rich as the training target for the autoregressive language model. Subsequently, a non-autoregressive model is employed to predict discrete acoustic codecs that contain fine-grained acoustic details. The TTS model focuses solely on linguistic modeling during autoregressive training, thereby reducing the error propagation that occurs in non-autoregressive training. Both objective and subjective evaluations validate the effectiveness of our proposed method.
Index Terms: language models, text-to-speech

1. Introduction

Text-to-Speech (TTS) aims to produce natural and intelligible speech from text input [1]. With the advent of deep learning methods, neural network-based TTS systems have significantly enhanced the quality of synthesized speech [2]. TTS has become a pivotal technology in conversational artificial intelligence [3] and human-machine interaction [4].

The primary challenge for text-to-speech systems is the computational modeling of the one-to-many mapping function between text and speech [1]. A TTS model is expected to synthesize highly natural speech across various styles for multiple speakers, including unseen speakers or styles (zero-shot TTS) [2]. Previous TTS studies [5, 6, 7, 8] have largely focused on designing a speaker or reference encoder to capture speaker characteristics and incorporate them into a TTS framework. Since speaker characteristics are closely entangled with speech content and style [9, 10], speaker similarity and speech naturalness often decline in zero-shot scenarios [6]. Moreover, these frameworks require high-quality recorded speech data for training, which constrains the diversity of the synthesized speech.

Recent language models have achieved great success in natural language generation [11, 12] and then applied to audio generation [13] and speech synthesis [14, 15]. Language model (LM) based TTS systems [16, 17] formulate text-to-speech as a conditional language modeling task, where the speech waveform is quantized into discrete tokens and is then modeled with a language model. Compared to conventional neural TTS, language model-based TTS systems have demonstrated strong in-context learning capabilities on large and diverse speech data, and can synthesize highly natural speech. Despite their advancements, these systems often face issues with robustness and suffer from word deletion, repetition, and mispronunciation [16], which makes them less ideal for real-life applications.

One notable challenge in text-to-speech systems based on language models lies in their autoregressive approach. For instance, in [16], text tokens are mapped to speech tokens in an autoregressive manner. However, accurately modeling speech tokens is difficult due to the presence of both verbal and non-verbal information in the speech signal [18, 19]. Furthermore, additional challenges arise from confounding factors such as environmental noises, which are encoded into speech tokens during quantization [20]. False predictions of speech tokens can accumulate, leading to robustness issues in synthesized speech. In this paper, we introduce a phonetic enhanced language modeling method for TTS that leverages self-supervised learning (SSL) representations [21] as the training target for autoregressive training. This approach allows the TTS framework to focus solely on linguistic modeling during autoregressive training. Subsequently, a non-autoregressive model is employed to recover fine-grained acoustic details of speech from the predicted SSL representations. Our contributions are summarized as follows:

- We propose a phonetic enhanced language modeling method to improve the performance of language model-based TTS;
- We leverage SSL representations, which are phonetically rich, as the training target for autoregressive language model training. We map text tokens to phonetic variations autoregressively, and phonetic variations to acoustic details in a non-autoregressive manner, thereby reducing the error propagation;
- Our method enhances the performance of TTS frameworks by effectively improving system robustness and achieving better speech quality in terms of naturalness and zero-shot speaker similarity compared to the baseline.

The rest of the paper is organized as follows: We introduce the related work on language model-based TTS in Section 2. Our proposed method is introduced in Section 3. In Section 4, we report our experimental results. Section 5 concludes the paper.

2. Language-Model-Based TTS

Text-to-Speech (TTS) models facilitate the translation between text and speech. TTS systems, such as Tacotron [22] and Transformer-based TTS [23, 24], utilize sequence-to-sequence autoregressive models, significantly improving the naturalness of synthesized voices. However, the output sequences of TTS are continuous and often longer than text inputs, making it challenging to model abundant speech variations from discrete text inputs [25]. Incorrect predictions of text-acoustic alignment of-

10.21437/Interspeech.2024-359

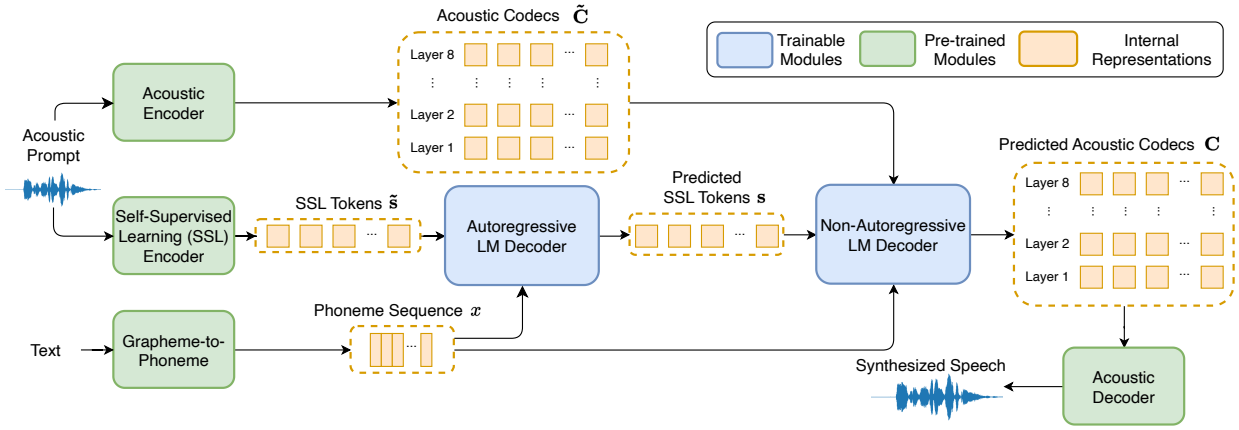


Figure 1: *The overall diagram of proposed phonetic enhanced language model (LM) based text-to-speech framework. Given input text and an acoustic prompt, the autoregressive decoder predicts self-supervised learning (SSL) tokens that contain phonetic information, and the non-autoregressive decoder further predicts 8 layers of acoustic codecs that represent fine-grained acoustic details.*

ten result in robustness issues for synthesized voices [26]. Non-autoregressive TTS models, such as FastSpeech [27] and Glow-TTS [28], introduce a separate duration predictor, which improves TTS robustness but at the cost of reduced speech diversity and expressiveness [2].

Inspired by the recent successes of prompting-based language models [11], language model-based TTS [16, 15, 29] has garnered attention. Specifically, neural audio codecs [20, 30] facilitate the integration of speech units into a language model by quantizing speech waveforms into highly compressed audio codecs. TTS is then formulated as a task of conditional language modeling rather than as traditional signal regression. VALL-E [16] represents one successful example of language model-based TTS, which demonstrates scalability and the capability for in-context learning. It employs a two-stage training process: modeling the first layer of audio codecs autoregressively and predicting the following layers in parallel. VALL-E can be trained on large and diverse multi-speaker datasets and clone the voices of speakers from just a short acoustic prompt. However, VALL-E inherits synthesis robustness issues, such as word mispronunciation, deletion, and repetition, from autoregressive language models [16].

In this paper, we propose a phonetic enhanced language modeling method, aiming to facilitate a more robust language modeling for speech units. The most related study, SPEAR-TTS [15], decouples TTS into two sequence-to-sequence tasks: from text to semantic tokens ('Reading') and from semantic tokens to speech waveforms ('Speaking'). The primary objective of SPEAR-TTS is to reduce the need for paired text and audio data, while our objective is to improve the robustness of language model-based TTS systems.

3. Phonetic Enhanced Language Modeling

In this section, we introduce our proposed phonetic enhanced language model (LM) based text-to-speech (TTS) system. Figure 1 illustrates the overall diagram. The overall diagram follows the pipeline of "Phoneme → Phonetic → Acoustic". Given phoneme inputs and an acoustic prompt as inputs, the autoregressive decoder first predicts SSL tokens that contain phonetic information. The non-autoregressive decoder then predicts 8 layers of acoustic codecs that represent fine-grained acoustic details. Finally, from these predicted acoustic codecs, the acoustic decoder synthesizes the speech waveform.

3.1. Problem Formulation

Following VALL-E [16], we regard zero-shot TTS as a conditional language modeling task. Given phoneme sequences and an acoustic prompt, the TTS model is trained to generate discrete acoustic codecs. These codecs capture all the necessary acoustic details for synthesizing a speech waveform, which not only represent the linguistic content of the phoneme sequence but also reflect the speaker's characteristics as indicated by the acoustic prompt. Encodec [20] is often used to quantize acoustic prompt into 8 layers of discrete acoustic codecs. The first layer of codecs is predicted autoregressively, while the remaining layers are generated in parallel, based on the first layer of codecs. However, given the paramount importance of the first quantizer in the reconstruction process [20, 30], the autoregressive model is required to process an excessive amount of information. Consequently, this could lead to an accumulation of errors and result in robustness issues in synthesized speech. In response to this challenge, we propose a phonetic enhanced language model approach to help with the autoregressive training, which we will introduce next.

3.2. Phonetic Enhanced Autoregressive Language Modeling

Rather than directly mapping phoneme sequence to acoustic tokens, we propose an approach that maps phoneme sequence to phonetic tokens in an autoregressive manner, which we refer to as "phonetic enhanced". Drawing inspiration from the success of self-supervised learning (SSL) representations in automatic speech recognition, we leverage SSL representations as phonetically rich targets for autoregressive training. The phonetic richness of SSL features arises from the models' training objective to capture the underlying structure of the audio signal [21]. For instance, to accurately predict masked parts of speech, a SSL model requires a deep understanding of phonetics and the combinations of sounds in natural language [31]. Moreover, these models are trained on large and diverse datasets of speech, exposing them to a wide range of phonetic variations, including different accents and speaking styles. This exposure

allows them to learn robust phonetic representations across various speakers and contexts [32].

We train the autoregressive language model to predict the SSL tokens from the phoneme sequences. Given a dataset $D = \{x_i, y_i\}$, where y_i is the i-th audio sample and x_i is the corresponding phoneme sequence, we train an autoregressive (AR) decoder-only language model to predict SSL tokens $\mathbf{s}$, which is conditioned on the phoneme sequence $\mathbf{x}$ and the SSL tokens $\tilde{\mathbf{s}}$ extracted from the acoustic prompt:

$$p(\mathbf{s}|\mathbf{x}, \tilde{\mathbf{s}}; \theta_{AR}) = \prod_{t=0}^{T} p(\mathbf{s}_t|\mathbf{s}_{<t}, \tilde{\mathbf{s}}, \mathbf{x}; \theta_{AR}) \quad (1)$$

where T is the sequence length of SSL tokens $\tilde{\mathbf{s}}$.

The AR model aims to predict SSL tokens from the input phoneme sequence. We refer this stage as "Phoneme $\rightarrow$ Phonetic". Since SSL tokens not only encode contextual cues but also prosodic variations, the AR model learns to map phoneme sequence to a wide range of phonetic variations.

3.3. Non-Autoregressive Language Modeling

The non-autoregressive (NAR) language model is trained to recover fine-grained acoustic details from phonetic variations, a stage we refer to as "Phonetic $\rightarrow$ Acoustic". The NAR model is conditioned on the phoneme sequence $\mathbf{x}$, the predicted SSL tokens $\mathbf{s}$, and the acoustic codecs $\tilde{\mathbf{C}}$ extracted from the acoustic prompt:

$$p(\mathbf{C}|\mathbf{x}, \tilde{\mathbf{C}}; \theta_{NAR}) = \prod_{j=1}^{8} p(\mathbf{c}_{:,j}|\mathbf{C}_{:,<j}, \mathbf{s}, \tilde{\mathbf{C}}, \mathbf{x}; \theta_{NAR}) \quad (2)$$

The acoustic codecs $\mathbf{C}$ consist of 8 layers of discrete codec sequences, each containing fine-grained acoustic details that could be used to reconstruct a speech waveform. The amount of information decreases with each layer.

3.4. Inference via Prompting

During inference, an acoustic prompt is provided, which triggers the SSL encoder and acoustic encoder to generate SSL tokens and 8 layers of acoustic codecs, respectively. Subsequently, the AR decoder predicts SSL tokens from the input phoneme sequence and the SSL tokens. Then, the NAR decoder predicts acoustic codecs using the predicted SSL tokens, acoustic codecs from the acoustic prompt, and the phoneme sequence. Finally, the acoustic decoder synthesizes the speech waveform based on the predicted acoustic codecs.

4. Experiments

4.1. Experimental Setup

We perform all experiments on the LibriTTS dataset [33], which consists of around 600 hours of speech data with transcriptions at a 24kHz sampling rate from 2,456 speakers. We merge the 'train-clean' and 'train-other' subsets to train the TTS framework and evaluate its performance on the 'test-clean' and 'test-other' subsets, respectively. Note that the 'test-other' subset contains more challenging recording conditions compared to the 'test-clean'. The speakers from both 'test-other' and 'test-clean' subsets are unseen during training.

We evaluate the zero-shot performance of our TTS framework on the test sets of the LibriTTS dataset, where the framework is expected to synthesize speech mimicking the voice of a given speaker not seen during training. To set up the experiments, for each speaker in the test set, we randomly choose one speech sample from the same speaker to serve as the acoustic prompt, distinct from the speech to be synthesized, and then synthesize the speech based on the transcriptions. All the synthesized speech is within 3 to 10 seconds. All the demos are publicly available online[1].

For comparison, we choose VALL-E [16] as the baseline framework. VALL-E utilizes a similar decoder-only language model (LM) and predicts the first layer of acoustic codecs autoregressively, followed by predicting the other layers non-autoregressively. Additionally, we compare different configurations using self-supervised learning (SSL) tokens. Specifically, we compare Hubert [34] and WavLM [35] features, followed by K-means clustering [36] with the cluster sizes of 500 and 1024 for both Hubert and WavLM.

4.2. Implementation Details

We implemented VALL-E based on an unofficial release[2]. Our proposed framework shares a similar architecture with VALL-E. Both the autoregressive and non-autoregressive models consist of a transformer architecture consisting of 12 layers, 16 attention heads, an embedding dimension of 1024, a feed-forward layer dimension of 4096 and a dropout rate of 0.1. We use a 24kHz Encodec[3] as the acoustic encoder to compress the speech waveform into 8 layers of discrete acoustic codes, while the Encodec decoder is utilized to recover 8 layers of discrete acoustic codes to the speech waveform. To extract SSL tokens from the speech waveform, we use official versions of the Hubert[4] and WavLM[5] models. We obtain the features from the 6th Hubert layer and the 24th WavLM layer respectively, and cluster them using the K-means algorithm [36]. We manually upsample the SSL tokens from 16kHz to 24kHz to align with acoustic codecs. We train the TTS framework on 4 NVIDIA TESLA V100 64GB GPUs for 1M steps. The other training configurations remain the same as those in the unofficial release[2] of VALL-E.

4.3. Objective Evaluation

We conduct objective evaluation to assess the robustness performance of our TTS framework. We use Whisper[6] to generate the transcriptions from the synthesized speech and calculate word error rate (WER) and character error rate (CER) with the ground-truth transcriptions.

We have reported the objective results in Table 1. Our proposed phonetic enhanced LM-based TTS framework consistently outperforms the baseline VALL-E on both 'test-clean' and 'test-other' on WER and CER metrics. Regarding different SSL configurations, we observe that the proposed framework with WavLM tokens clustered with a K-means cluster size of 1024 ('with WavLM (K = 1024)') achieves the best performance on both 'test-clean' and 'test-other', while WavLM with a cluster size of 500 ('with WavLM (K = 500)') performs slightly worse than the others. Compared to WavLM, the cluster size of Hubert features does not have a significant impact on the final results. However, their performance declines on the

[1] **Speech Demos**: https://demos46.github.io/IS24/

[2] https://github.com/lifeiteng/vall-e

[3] https://huggingface.co/facebook/encodec_24khz

[4] https://dl.fbaipublicfiles.com/hubert/hubert_xtralarge_ll60k_finetune_ls960.pt

[5] https://huggingface.co/microsoft/wavlm-large

[6] https://huggingface.co/openai/whisper-large-v3

Table 1: *Objective evaluation results to compare the robustness of the baseline VALL-E and our proposed TTS systems with different configurations using SSL tokens. 'WER' and 'CER' denote as 'Word Error Rate' and 'Character Error Rate' respectively, where a smaller value indicates a better system robustness. 'K' represents the K-means cluster size of the SSL tokens.*

Test Dataset		LibriTTS (test-clean)		LibriTTS (test-other)	
Objective Metrics (%)		WER↓	CER↓	WER↓	CER↓
Baseline (VALL-E)		21.05	18.19	25.56	23.51
Proposed Phonetic Enhanced LM-based TTS	w/ Hubert (K = 500)	8.03	7.18	13.39	11.58
	w/ Hubert (K = 1024)	7.66	5.84	12.14	10.05
	w/ WavLM (K = 500)	12.72	11.96	14.34	12.73
	w/ WavLM (K = 1024)	6.12	5.44	7.92	6.11

Table 2: *Subjective evaluation results with 95% confidence interval to assess the naturalness ('MOS') and the speaker similarity ('S-MOS') of the baseline VALL-E and our proposed TTS systems with different configurations using SSL tokens.*

Test Dataset		LibriTTS (test-clean)		LibriTTS (test-other)	
Subjective Metrics		MOS	S-MOS	MOS	S-MOS
Reference Speech		4.81±0.07	−*	4.39±0.11	−*
Baseline (VALL-E)		2.51±0.19	3.12±0.17	2.49±0.16	3.09±0.13
Proposed Phonetic Enhanced LM-based TTS	w/ Hubert (K = 500)	3.58±0.13	3.89±0.13	3.43±0.13	3.35±0.14
	w/ Hubert (K = 1024)	3.29±0.14	3.59±0.14	3.19±0.10	3.33±0.15
	w/ WavLM (K = 500)	3.38±0.13	3.61±0.16	3.33±0.14	3.53±0.13
	w/ WavLM (K = 1024)	3.54±0.13	3.94±0.13	3.53±0.10	3.55±0.15

* S-MOS results for 'Reference Speech' are not reported because it serves as the comparison target in speaker similarity evaluation.

'test-other', indicating that Hubert tokens are less noise-robust than WavLM tokens.

4.4. Subjective Evaluation

We further conduct two listening tests to assess the performance of our TTS framework. In each test, 10 native English speakers participated, with each participant listening to a total of 200 synthesized speech samples (200 samples = 10 samples × 5 systems × 2 test sets × 2 tests). The first listening test (referred as 'MOS') evaluates the naturalness of the synthesized speech samples. Participants are asked to score each sample on a 5-point scale (5: Excellent, 4: Good, 3: Satisfactory, 2: Unsatisfactory, 1: Bad). The second listening test (referred as 'S-MOS') evaluates the speaker similarity with the reference speech. Participants are asked to score each sample on a 5-point scale (5: Very Similar, 4: Similar, 3: Neutral, 2: Dissimilar, 1: Very Dissimilar). The results are reported in Table 2.

We first observe that all of our proposed frameworks significantly outperform the Baseline (VALL-E) in naturalness and speaker similarity on both test sets ('test-clean' and 'test-other'). This demonstrates that our phonetic enhanced language modeling approach improves the robustness of the TTS system and contributes to better zero-shot performance for synthesized speech.

We further investigate the impacts of different configurations of SSL tokens. From Table 2, we observe that the proposed framework with Hubert features clustered with a size of 500 ('w/ Hubert (K=500)') outperforms that with a size of 1024 ('w/ Hubert (K=1024)') on the test set of LibriTTS (test-clean). Conversely, the proposed framework with WavLM features clustered with a size of 1024 ('w/ WavLM (K=1024)') outperforms that with a size of 500 ('w/ WavLM (K=500)') on the same test set. Additionally, we note that the proposed ('w/ WavLM (K=1024)') achieves slightly better speaker similarity than the proposed ('w/ Hubert (K=500)'), while the naturalness slightly declines.

We observe that all frameworks suffers from a performance decline on the test set of LibriTTS (test-other), which contains more challenging recording conditions. Notebly, the performance of the proposed frameworks with WavLM tokens presents a more noise-robust performance by achieving less performance decline than those with Hubert tokens.

4.5. Discussions

The major difference between our proposed TTS framework and the baseline VALL-E is the utilization of self-supervised representations as phonetic targets for autoregressive language modeling. Both objective and subjective evaluations reveal two key findings:

- The proposed TTS framework utilizing WavLM tokens demonstrates greater noise robustness compared to those using HuBERT tokens;
- A larger cluster size in K-means clustering of WavLM tokens contributes to improved performance.

As a result, configurations using WavLM features with a larger cluster size outperform other setups. This advantage is largely credited to WavLM's pre-training strategy, which includes denoising and handling distorted audio signals, thereby enhancing its ability to capture phonetic details under challenging conditions. WavLM leverages more detailed phonetic information provided by a larger cluster and better model the nuances in complex audio environments. However, increasing the cluster size in Hubert could potentially add complexity to non-autoregressive acoustic modeling and might lead to diminished performance.

Our future improvement focuses on the acoustic encoder, for which we use Encodec [20] in our experiments. We found that the performance of our proposed TTS system is primarily limited by the effectiveness of Encodec, particularly in terms of its reconstruction distortion, noise robustness, codebook size, and compression bitrate.

5. Conclusion

This paper proposes a phonetic enhanced language modeling method for text-to-speech synthesis. By leveraging self-supervised representations as the training target for the autoregressive language model and predicting fine-grained acoustic details non-autoregressively, the text-to-speech framework learns to map phoneme sequences to phonetic variations and then to the acoustic information. Our experiments demonstrate that the proposed method could effectively reduce error propagation in speech unit modeling and improve the robustness and naturalness of the synthesized speech. Our experiments also compare different configurations of self-supervised representations and analyze their impacts on the synthesized results.

6. References

[1] T. Dutoit, *An introduction to text-to-speech synthesis*. Springer Science & Business Media, 1997, vol. 3.

[2] X. Tan, T. Qin, F. Soong, and T.-Y. Liu, "A survey on neural speech synthesis," *arXiv preprint arXiv:2106.15561*, 2021.

[3] J. Gao, M. Galley, and L. Li, "Neural approaches to conversational ai," in *The 41st international ACM SIGIR conference on research & development in information retrieval*, 2018, pp. 1371–1374.

[4] K. Zhou, B. Sisman, R. Rana, B. W. Schuller, and H. Li, "Speech synthesis with mixed emotions," *IEEE Transactions on Affective Computing*, vol. 14, no. 4, pp. 3120–3134, 2023.

[5] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. Lopez Moreno, Y. Wu *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," *Advances in neural information processing systems*, vol. 31, 2018.

[6] E. Casanova, J. Weber, C. D. Shulby, A. C. Junior, E. Gölge, and M. A. Ponti, "Yourtts: Towards zero-shot multi-speaker tts and zero-shot voice conversion for everyone," in *International Conference on Machine Learning*. PMLR, 2022, pp. 2709–2720.

[7] E. Cooper, C.-I. Lai, Y. Yasuda, F. Fang, X. Wang, N. Chen, and J. Yamagishi, "Zero-shot multi-speaker text-to-speech with state-of-the-art neural speaker embeddings," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6184–6188.

[8] Y. Wu, X. Tan, B. Li, L. He, S. Zhao, R. Song, T. Qin, and T.-Y. Liu, "Adaspeech 4: Adaptive text to speech in zero-shot scenarios," *arXiv preprint arXiv:2204.00436*, 2022.

[9] K. Zhou, B. Sisman, and H. Li, "Vaw-gan for disentanglement and recomposition of emotional elements in speech," in *2021 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2021, pp. 415–422.

[10] K. Zhou, B. Sisman, R. Rana, B. W. Schuller, and H. Li, "Emotion intensity and its control for emotional voice conversion," *IEEE Transactions on Affective Computing*, vol. 14, no. 1, pp. 31–48, 2023.

[11] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[12] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang *et al.*, "A survey on evaluation of large language models," *ACM Transactions on Intelligent Systems and Technology*, 2023.

[13] F. Kreuk, G. Synnaeve, A. Polyak, U. Singer, A. Défossez, J. Copet, D. Parikh, Y. Taigman, and Y. Adi, "Audiogen: Textually guided audio generation," *arXiv preprint arXiv:2209.15352*, 2022.

[14] Z. Borsos, R. Marinier, D. Vincent, E. Kharitonov, O. Pietquin, M. Sharifi, D. Roblek, O. Teboul, D. Grangier, M. Tagliasacchi *et al.*, "Audiolm: a language modeling approach to audio generation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.

[15] E. Kharitonov, D. Vincent, Z. Borsos, R. Marinier, S. Girgin, O. Pietquin, M. Sharifi, M. Tagliasacchi, and N. Zeghidour, "Speak, read and prompt: High-fidelity text-to-speech with minimal supervision," *arXiv preprint arXiv:2302.03540*, 2023.

[16] 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.

[17] Z. Zhang, L. Zhou, C. Wang, S. Chen, Y. Wu, S. Liu, Z. Chen, Y. Liu, H. Wang, J. Li *et al.*, "Speak foreign languages with your own voice: Cross-lingual neural codec language modeling," *arXiv preprint arXiv:2303.03926*, 2023.

[18] K. Zhou, B. Sisman, R. Liu, and H. Li, "Emotional voice conversion: Theory, databases and esd," *Speech Communication*, vol. 137, pp. 1–18, 2022.

[19] K. Zhou, B. Sisman, M. Zhang, and H. Li, "Converting Anyone's Emotion: Towards Speaker-Independent Emotional Voice Conversion," in *Proc. Interspeech 2020*, 2020, pp. 3416–3420.

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

[21] A. Mohamed, H.-y. Lee, L. Borgholt, J. D. Havtorn, J. Edin, C. Igel, K. Kirchhoff, S.-W. Li, K. Livescu, L. Maaløe *et al.*, "Self-supervised speech representation learning: A review," *IEEE Journal of Selected Topics in Signal Processing*, 2022.

[22] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio *et al.*, "Tacotron: Towards end-to-end speech synthesis," *arXiv preprint arXiv:1703.10135*, 2017.

[23] 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.

[24] T. Okamoto, T. Toda, Y. Shiga, and H. Kawai, "Transformer-based text-to-speech with weighted forced attention," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6729–6733.

[25] S. Lei, Y. Zhou, L. Chen, D. Luo, Z. Wu, X. Wu, S. Kang, T. Jiang, Y. Zhou, Y. Han *et al.*, "Improving language model-based zero-shot text-to-speech synthesis with multi-scale acoustic prompts," *arXiv preprint arXiv:2309.11977*, 2023.

[26] T. Okamoto, T. Toda, Y. Shiga, and H. Kawai, "Tacotron-based acoustic model using phoneme alignment for practical neural text-to-speech systems," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 214–221.

[27] 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.

[28] 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.

[29] K. Shen, Z. Ju, X. Tan, Y. Liu, Y. Leng, L. He, T. Qin, S. Zhao, and J. Bian, "Naturalspeech 2: Latent diffusion models are natural and zero-shot speech and singing synthesizers," *arXiv preprint arXiv:2304.09116*, 2023.

[30] 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.

[31] D. Wells, H. Tang, and K. Richmond, "Phonetic analysis of self-supervised representations of english speech," in *INTERSPEECH 2022*, 2022.

[32] M. de Seyssel, M. Lavechin, Y. Adi, E. Dupoux, and G. Wisniewski, "Probing phoneme, language and speaker information in unsupervised speech representations," in *Interspeech 2022*, 2022.

[33] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, "Libritts: A corpus derived from librispeech for text-to-speech," *Interspeech 2019*, 2019.

[34] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[35] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[36] S. Lloyd, "Least squares quantization in pcm," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.