MELA-TTS: JOINT TRANSFORMER-DIFFUSION MODEL WITH REPRESENTATION ALIGNMENT FOR SPEECH SYNTHESIS

Keyu An*, *Zhiyu Zhang**†, *Changfeng Gao**, *Yabin Li**, *Zhendong Peng**,
Haoxu Wang*, *Zhihao Du**, *Han Zhao**, *Zhifu Gao**, *Xiangang Li**

* Alibaba group † National Mobile Communications Research Laboratory, Southeast University
ankeyu.aky@alibaba-inc.com, zhiyuzhang@seu.edu.cn

ABSTRACT

This work introduces MELA-TTS, a novel joint transformer-diffusion framework for end-to-end text-to-speech synthesis. By autoregressively generating continuous mel-spectrogram frames from linguistic and speaker conditions, our architecture eliminates the need for speech tokenization and multi-stage processing pipelines. To address the inherent difficulties of modeling continuous features, we propose a representation alignment module that aligns output representations of the transformer decoder with semantic embeddings from a pretrained ASR encoder during training. This mechanism not only speeds up training convergence, but also enhances cross-modal coherence between the textual and acoustic domains. Comprehensive experiments demonstrate that MELA-TTS achieves state-of-the-art performance across multiple evaluation metrics while maintaining robust zero-shot voice cloning capabilities, in both offline and streaming synthesis modes. Our results establish a new benchmark for continuous feature generation approaches in TTS, offering a compelling alternative to discrete-token-based paradigms.

Index Terms— Transformer, diffusion, TTS, representation alignment.

1. INTRODUCTION

Autoregressive modeling based on discrete tokens has demonstrated remarkable success in text-to-speech (TTS) synthesis. Such frameworks critically depend on a pre-trained tokenizer to discretize continuous speech features into token sequences [1, 2]. During the generation process, an autoregressive model first performs next-token prediction, after which a dedicated decoder network maps the discrete tokens back to high-dimensional continuous speech features. While demonstrating exceptional proficiency in achieving high-fidelity speech naturalness and cross-speaker generalization through zero-shot voice cloning, this framework exhibits inherent limitations. First, the discretization of speech signals inherently incurs information loss, which fundamentally constrains the fidelity of subsequent speech reconstruction. Second and critically, the decoupled two-stage framework increases system complexity while creating a cascading error accumulation.

Recent studies have proposed end-to-end frameworks that directly generate continuous speech features without relying on discrete token intermediates [3, 4]. This paradigm shift eliminates the need for multi-stage pipelines while preserving the full information of raw speech features. However, these architectures still face critical challenges. Firstly, their performance lags behind state-of-the-art discrete-token-based models, particularly in content consistency [5]. Recently proposed DiTAR [4] achieves remarkable results in terms of low WER/CER on benchmarks. However, it's not clear whether it's robust on hard cases, such as long text containing repetitions, tongue twisters, and so on. Secondly, end-to-end architectures introduce significant optimization challenges: autoregressive modeling of continuous features typically requires substantially more training iterations to converge, compared to discrete-token-based autoregressive frameworks, due to the inherent complications in modeling high-dimensional continuous features.

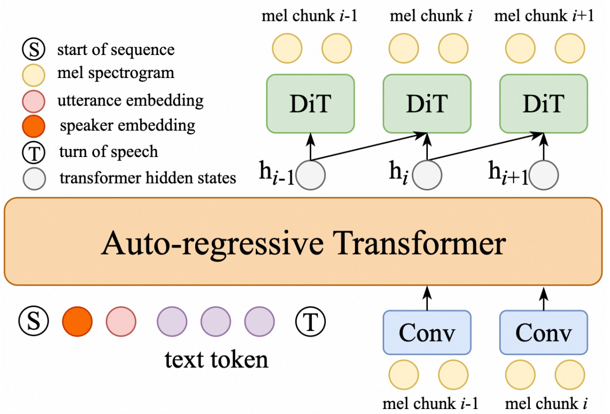

Fig. 1. The joint transformer and diffusion architecture. The autoregressive transformer decoder generates continuous vectors $\mathbf{h}$ as the condition to the diffusion model to generate the mel-spectrogram chunk.

To address these challenges, we propose MELA-TTS[1], a joint transformer and diffusion model that generates mel-spectrogram auto-regressively, eliminating the need for a speech tokenizer and multi-stage training and inference pipelines. To enhance content consistency and facilitate training convergence, we propose a representation alignment module that aligns the model's intermediate representations with those extracted from a pre-trained ASR encoder. The efficacy of the proposed method is validated through comprehensive experiments in two scenarios: (1) **offline synthesis**, which requires a complete text as input, and (2) **streaming synthesis**, where the input text is received in a streaming manner rather than given as a complete sentence in advance. Moreover, the scalability of MELA-TTS is evidenced by remarkable performance improvement when the training data scales to 170,000 hours.

The first two authors contribute equally to this work.

[1] MELA is the combination of MEL and Alignment.

2. METHODS

As illustrated in the Figure 1, MELA-TTS comprises an autoregressive transformer decoder and a diffusion module. The autoregressive transformer decoder generates continuous vectors $\mathbf{h}$ sequentially, and the diffusion module utilizes these vectors, along with speaker embeddings and utterance embeddings as conditional inputs, to perform a denoising process on the noisy mel-spectrogram chunk. Once the mel-spectrogram is generated, the speech waveform can be constructed using a neural vocoder. More importantly, we introduced a representation alignment module to align the continuous vectors $\mathbf{h}$ with the output representations of a pretrained ASR encoder, which encourages $\mathbf{h}$ to be more semantically informative, thereby improving the content consistency of the generated outputs. The detailed descriptions of each module are presented below.

2.1. Transformer decoder for auto-regressive modeling

In MELA-TTS, a transformer decoder autoregressively generates continuous vectors $\mathbf{h}$ conditioned on the utterance embedding, the speaker embedding, the tokenized text, and the mel-spectrogram history $\mathbf{X} = [x_1, x_2, ..., x_L]$. During training, the utterance embedding is extracted from randomly cropped segments of the input speech via a transformer encoder. The transformer encoder outputs features that are pooled into an utterance embedding vector, and is jointly optimized with the transformer decoder and the diffusion model. The speaker embedding is captured from the input speech with a pretrained speaker encoder [2]. During inference, both the utterance embedding and speaker embedding are derived from the prompt speech. The text input is first tokenized into BPE tokens with the tokenizer from Qwen2, and then converted to embeddings using Qwen 2's text embedding layer. As for the mel-spectrogram history, the i-th chunk of mel-spectrogram $\mathbf{X}^{(i)} = [x_{i\times N+1}, ..., x_{(i+1)\times N}] \in \mathcal{R}^{N\times D_{\text{mel}}}$ is downsampled and projected into a tensor of shape $[1, D_{\text{trans}}]$ by a strided convolution layer, and then fed into the transformer decoder. Here N is the chunk size, D_{mel} is the dimension of the mel-spectrogram, and D_{trans} is the dimension of the transformer decoder. Following [6], the output of the final transformer decoder layer $\mathbf{h}$ will serve as the condition for the diffusion model.

Unlike discrete-token-based TTS systems that terminate generation via prediction of a special end-of-sequence (EOS) token, MELA-TTS employs a stop prediction module to determine the end of synthesize. This module functions as a binary classifier: it takes the continuous hidden representation sequence $\mathbf{h}$ as input and outputs a binary decision (0/1) at each step, where 0 signifies continuation and 1 indicates termination of the speech synthesis process. The module is trained using a binary cross-entropy (BCE) loss $\mathcal{L}_{\text{stop}}$.

2.2. Diffusion for mel-spectrogram generation

In MELA-TTS, the diffusion module, implemented as a diffusion transformer [7], predicts a chunk of mel-spectrogram $\mathbf{X}_0^{(i)} := \mathbf{X}^{(i)}$ based on $[h_{i-1}, h_i]$, speaker embeddings $\mathbf{v}$, utterance embeddings $\mathbf{u}$, and noisy mel-spectrogram chunk with previous mel-spectrogram chunk prepended $[\mathbf{X}_0^{(i-1)}, \mathbf{X}_t^{(i)}]$:

$$\begin{aligned}\hat{\mathbf{X}}_0^{(i)} &= \text{DiT}(\Psi_i, [\mathbf{X}_0^{(i-1)}, \mathbf{X}_t^{(i)}]) \\ &= \text{DiT}([h_{i-1}, h_i], \mathbf{v}, \mathbf{u}, [\mathbf{X}_0^{(i-1)}, \mathbf{X}_t^{(i)}]).\end{aligned}$$

Here $\Psi_i = \{[h_{i-1}, h_i], \mathbf{v}, \mathbf{u}\}$ is the condition, and $\mathbf{X}_t^{(i)} = \alpha_t \mathbf{X}_0^{(i)} + \sigma_t \epsilon$ is given by a diffusion forward process [8], and ϵ is the standard gaussian noise. We follow the variance preserving (VP) formulation and set $\alpha_t = \cos(\frac{\pi t}{2})$ and $\sigma_t = \sin(\frac{\pi t}{2})$. The previous continuous vector h_{i-1} and the mel-spectrogram chunk $\mathbf{X}_0^{(i-1)}$ are provided as prefix context for the diffusion model, and the output of the prefix part will be discarded. The loss is defined as the L2 distance between the predicted mel-spectrogram and the ground truth mel-spectrogram:

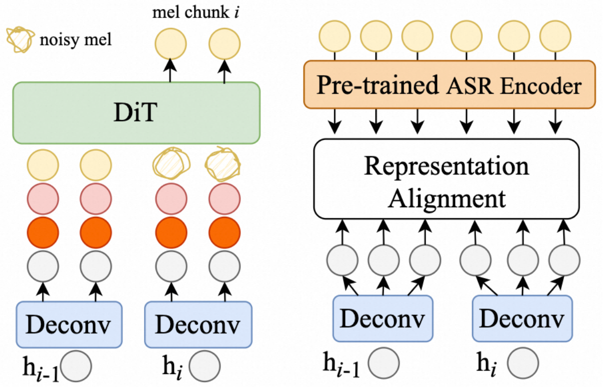

Fig. 2. Left: the diffusion module utilizes $\mathbf{h}$, along with speaker embeddings $\mathbf{v}$ and utterance embeddings $\mathbf{u}$ as conditional inputs, to perform mel-spectrogram denoising. $\mathbf{h}$, $\mathbf{v}$, and $\mathbf{u}$ are upsampled respectively to align with the chunk size of the mel-spectrogram. Right: the representation alignment module. $\mathbf{h}$ is also upsampled to align with the length of the pretrained semantic representation.

$$\mathcal{L}_{\text{diff}} = \sum_i (\hat{\mathbf{X}}_0^{(i)} - \mathbf{X}_0^{(i)})^2.$$

2.3. Representation alignment module

In discrete-token-based TTS systems, supervised semantic tokens, which are typically derived from an ASR model, have demonstrated superior efficacy as intermediate representations, significantly improving content consistency and voice cloning performance [9]. However, in end-to-end models, since the model directly predicts mel-spectrograms or other continuous representations, it is not explicitly guided to produce semantically enriched intermediates. The absence of intermediate semantic guidance leads to two adverse consequences: poor content consistency in the synthesized speech, and slower convergence during model training. To address it, we propose a representation alignment module, as illustrated in Figure 2. Specifically, we align the output of the autoregressive transformer $\mathbf{h}$ with pretrained semantic representations $\mathbf{h}_{\mathbf{asr}}$ generated by an ASR encoder by adding a cosine similarity loss term between them:

$$\mathcal{L}_{\text{align}} = \text{CosineSimilarity}(\text{TAM}(\mathbf{h}), \mathbf{h}_{\mathbf{asr}}),$$

where TAM is a time alignment module to resolve temporal resolution mismatches between $\mathbf{h}$ and $\mathbf{h}_{\mathbf{asr}}$, implemented as a linear layer followed by reshape operations.

For alignment objective, one might intuitively consider using the mel-spectrogram directly as the alignment target. However, experimental results revealed that this strategy fails to provide any positive gains and instead significantly degrades both content consistency and speaker similarity in voice cloning. We will discuss it in section 3.2.

To sum up, the overall training loss is defined as:

$$\mathcal{L} = \mathcal{L}_{\text{diff}} + \mathcal{L}_{\text{stop}} + \mathcal{L}_{\text{align}}$$

[2] https://github.com/modelscope/3D-Speaker

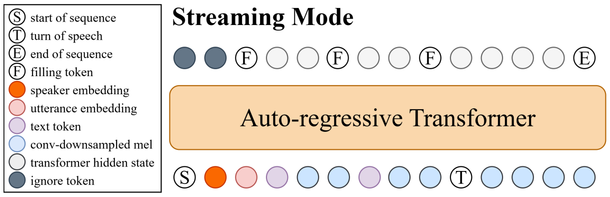

Fig. 3. A diagram of the auto-regressive language model for streaming synthesis in MELA-TTS.

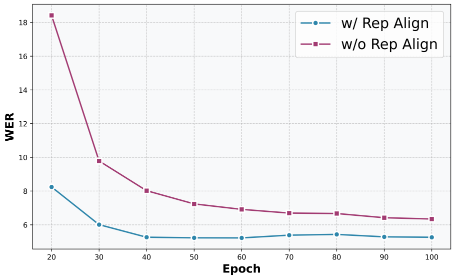

Fig. 4. Comparison of WER over training epochs with and without representation alignment.

2.4. Streaming synthesis

For streaming synthesis (Figure 3), we interleave the text tokens and continuous conv-downsampled mel-spectrogram in an $n : m$ ratio, which enables incremental speech synthesis and allows the generation of m mel-spectrogram chunks for every n text tokens received. The model is simultaneously trained on both interleaved and non-interleaved sequences, thus streaming and non-streaming synthesis can be performed within a unified model. The turn-of-speech token indicates the end of text input, and the filling token only marks the position and is excluded for target prediction and loss calculation. The termination of speech generation is determined by the binary classification module, which is the same as the offline model.

3. EXPERIMENTS

3.1. Experiment settings

3.1.1. Datasets

The experiments are conducted on 585-hour LibriTTS [14], and an in-house 170,000-hour dataset, including 130,000-hour Chinese, 30,000-hour English, and 10,000-hour other languages. Experiments on LibriTTS are mainly for ablation studies, and the dataset of 170,000 hours is used to evaluate the scaling ability.

Table 1. Ablation study of streaming synthesis, utterance embedding (Utt Emb), and representation alignment (Rep Align). $*$ indicates using mel-spectrogram instead of the pretrained ASR encoder output as the representation alignment target.

Exp ID	**Streaming**	**Utt Emb**	**Rep Align**	WER ↓	SS1 ↑	SS2 ↑
0	✗	✗	✗	6.3	0.46	0.55
1	✗	✗	✓	5.3	0.46	0.54
2	✗	✗	✓ $*$	6.7	0.41	0.48
3	✗	✓	✗	6.0	0.47	0.57
4	✗	✓	✓	5.2	**0.48**	**0.58**
5	✓	✗	✗	6.6	0.46	0.55
6	✓	✓	✓	**5.0**	**0.48**	**0.58**

3.1.2. Model configuration

In MELA-TTS, all waveforms are resampled at 24 kHz, and the feature is the 80-dimensional mel-spectrogram extracted at 50 Hz with a window length of 1920 and a hop length of 480. The default size of the mel chunk is 8 (160ms). Thus, the autoregressive transformer works at a rate of 6.25 Hz (50/8 Hz) to generate continuous vectors $\mathbf{h}$, much smaller than most discrete-token-based TTS systems (25Hz for CosyVoice3 and 75Hz for VALL-E). With the interleaving ratio $n : m$ set to 4:3, MELA-TTS generates 3 mel-spectrogram chunks (480ms of speech) for every 4 text tokens received.

The transformer decoder follows the configuration of the pre-trained textual LLM, Qwen2-0.5B [15], and is initialized using its weights. The diffusion module is a 22-layer diffusion transformer with 1024 hidden states and 16 heads. Following [16], the diffusion module is trained on both conditional and non-conditional situations to enable the classifier-free guidance (CFG) [17] at inference:

$$\hat{\mathbf{X}}_{0,\text{cfg}}^{(i)} = (1+\alpha)\text{DiT}(\Psi_i, [\mathbf{X}_0^{(i-1)}, \mathbf{X}_t^{(i)}]) - \alpha\text{DiT}(\emptyset, [\mathbf{X}_0^{(i-1)}, \mathbf{X}_t^{(i)}]),$$

and α is set to 0.7. For sampling, we use the DDIM sampler [18], which accelerates generation by adopting a deterministic sampling process, and the default number of function evaluations (NFE) is 10.

We adopt the encoder of SenseVoice-Large [19] to produce semantic representations for representation alignment. Note that the input feature of the pre-trained ASR encoder is not necessarily the same as the feature we adopted for TTS. For SenseVoice-Large, the input waveform is resampled at 16 kHz, and a 128-dimensional mel-spectrogram is computed with a window length of 400 and a hop length of 160. The encoder downsample the mel-spectrogram by a factor of 4, yielding an output representation $\mathbf{h}_{\mathbf{asr}}$ at 25Hz. Thus, the time alignment module (TAM) upsample $\mathbf{h}$ by a factor of 4 to match the temporal resolution of $\mathbf{h}_{\mathbf{asr}}$.

3.1.3. Metrics

We evaluate MELA-TTS's speech generation with CER/WER for content consistency, and cosine similarity between generated speech and reference speech for voice cloning speaker similarity (SS). Specifically, we use Whisper-large V3 [20] to calculate English WER and Paraformer [21] to calculate Chinese CER. SS is calculated on the speaker embedding extracted by WavLM-TDNN [22] (the result is denoted as SS1), or the ERes2Net speaker verification model [23] (denoted as SS2).

3.2. Ablation study on LibriTTS

We conducted ablation studies on LibriTTS and evaluated on seed-tts-eval test-en[12] to quantify the individual and combined contribution of utterance embedding and representation alignment. As presented in Table 1, in offline mode, the model without either component establishes a baseline of WER = 6.3, SS1 = 0.46, and SS2 = 0.55. Incorporating representation alignment alone yields a substantial 1.0-point reduction in WER (6.3 → 5.3). Moreover, as shown in Figure 4, representation alignment accelerates training by over 3.3×, reaching comparable performance of the model trained over 100 epochs without representation alignment, in less than 30 epochs. Speaker similarity (SS1 0.46 → 0.47, SS2 0.55 → 0.57) improves when utterance embedding is introduced, which demonstrates its capacity to enhance the modeling ability of speaker information. Most significantly, the system that combines utterance embedding and representation alignment achieves the optimal offline performance with

Table 2. Zero-shot TTS performance comparison between MELA-TTS and results from literature on seed-tts-eval. † indicates that the model is trained using the same data, so the results are comparable.

Model	***test-zh***			***test-en***			***test-hard***		
	CER ↓	**SS1 ↑**	**SS2 ↑**	**WER ↓**	**SS1 ↑**	**SS2 ↑**	**CER ↓**	**SS1 ↑**	**SS2 ↑**
Human	1.3	0.76	0.78	2.1	0.73	0.74	-	-	
Non-autoregressive Models									
F5-TTS [10]	1.6	0.74	0.80	1.8	0.65	0.74	8.7	0.71	0.76
MaskGCT [11]	2.3	0.77	0.75	2.6	0.71	0.73	10.3	0.75	0.72
Autoregressive Models									
Seed-TTS [12]	1.1	**0.80**	-	2.3	**0.76**	-	7.6	**0.78**	-
DiTAR [4]	1.0	0.75	-	**1.7**	0.74	-	-	-	-
CosyVoice [9] †	3.6	0.72	0.78	4.3	0.61	0.70	11.8	0.71	0.76
CosyVoice 2.0 [13] †	1.5	0.75	0.81	2.6	0.65	0.74	**6.8**	0.72	0.78
CosyVoice 3.0-0.5B [2] †	1.3	0.75	**0.81**	2.5	0.65	**0.75**	7.0	0.72	**0.79**
MELA-TTS									
w/o rep align †	1.2	0.74	0.79	4.0	0.60	0.68	10.9	0.72	0.78
w/ rep align †	**0.9**	0.72	0.77	2.4	0.59	0.68	7.6	0.71	0.76
streaming mode w/ rep align †	**0.9**	0.72	0.78	2.5	0.59	0.68	7.7	0.71	0.77

a WER of 5.2, SS1 of 0.48, and SS2 of 0.58, which shows clear synergy. We attribute this to the complementary roles of the two modules: representation alignment explicitly regularizes cross-modal semantic consistency, thereby freeing utterance embedding to specialize in fine-grained acoustic modeling, such as the speaker information. These findings underscore that jointly modeling utterance-level information and cross-modal alignment achieves a superior balance between content consistency and speaker similarity.

Directly using mel-spectrograms as the alignment target degrades both WER, SS1, and SS2 (Exp 2 vs. Exp 0), which suggests that aligning to pretrained ASR encoder representations is a more effective objective, presumably because it encourages a decoupled semantic-acoustic modeling: the autoregressive transformer produces semantically informative representations, and the acoustic details are reconstructed by the diffusion model. This is consistent with the findings in discrete-token-based TTS systems, where a decoupled semantic-acoustic modeling is proven to benefit both content consistency and voice cloning capability [9, 13].

Table 1 further compares the performance of streaming and offline synthesis. In streaming mode, MELA-TTS exhibits comparable WER to offline mode: 6.3 vs. 6.6 for the baseline condition, and 5.2 vs. 5.0 when both the representation alignment and utterance embedding modules are incorporated. Furthermore, both SS1 and SS2 remain nearly identical to those obtained under the offline configuration, which demonstrates great robustness of MELA-TTS in streaming mode.

3.3. Evaluation on large-scale data

Results on 170,000-hour data are presented in Table 2. For all experiments on 170,000-hour data, utterance embedding is adopted by default. Consistent with the findings on LibriTTS, the representation alignment module produces significant improvement on content consistency (25%, 40%, and 30% relative CER/WER reduction on test-zh, test-en, and test-hard, respectively), with little degradation on speaker similarity. Streaming synthesis performs equally well as the offline mode. Compared with the results on LibriTTS, data scaling yields substantial performance improvement (WER 5.2 → 2.4, SS1 0.48 → 0.59 and SS2 0.58 → 0.68 on test-en), which demonstrates the superior scaling ability of MELA-TTS.

When compared to other recently proposed models, MELA-TTS with representation alignment achieves state-of-the-art WER/CER results on test-zh, much better than the discrete-token based counterpart CosyVoice using the same training data, and is comparable with other competitive models on test-en and test-hard. Notably, while continuous representation-based DiTAR achieves the lowest WER on test-en, it's not been evaluated on test-hard, so it's not clear whether it's robust enough on hard cases, e.g. generating long utterances with challenging patterns for autoregressive models, such as word repetitions, tongue twisters, and so on.

For voice cloning speaker similarity, MELA-TTS with representation is comparable with competitive models on tesh-zh and test-hard, but lags in test-en. A possible reason for the suboptimal performance on speaker similarity is that in MELA-TTS, the diffusion module can only leverage the local context, while in discrete-token-based multi-stage system, like CosyVoice series, the diffusion or flow-matching module can utilize all input tokens (all history tokens in the streaming mode), as well as the prompt speech, as conditions to generate the mel-spectrogram. Similar speaker similarity gaps have also been observed in other continuous representations-based systems [3, 24]. We leave the optimization of the voice cloning ability for future work.

4. CONCLUSIONS

We propose MELA-TTS, a joint transformer-diffusion framework for end-to-end text-to-speech synthesis, eliminating the dependency on speech tokenization and multi-stage processing pipelines. We further propose a representation alignment module to enhance the model's ability to capture semantic information. The proposed model is evaluated in both non-streaming and streaming modes, on datasets with scales varying from 585 to over 170,000 hours, demonstrating its effectiveness. In the future, we will further enhance the voice cloning capability of MELA-TTS and explore its applications in other domains, such as audio and music generation.

5. REFERENCES

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

[2] Zhihao Du, Changfeng Gao, Yuxuan Wang, Fan Yu, Tianyu Zhao, Hao Wang, Xiang Lv, Hui Wang, Chongjia Ni, Xian Shi, et al., "Cosyvoice 3: Towards in-the-wild speech generation via scaling-up and post-training," *arXiv preprint arXiv:2505.17589*, 2025.

[3] Lingwei Meng, Long Zhou, Shujie Liu, Sanyuan Chen, Bing Han, Shujie Hu, Yanqing Liu, Jinyu Li, Sheng Zhao, Xixin Wu, Helen Meng, and Furu Wei, "Autoregressive speech synthesis without vector quantization," 2025.

[4] Dongya Jia, Zhuo Chen, Jiawei Chen, Chenpeng Du, Jian Wu, Jian Cong, Xiaobin Zhuang, Chumin Li, Zhen Wei, Yuping Wang, and Yuxuan Wang, "Ditar: Diffusion transformer autoregressive modeling for speech generation," 2025.

[5] Xinfa Zhu, Wenjie Tian, and Lei Xie, "Autoregressive speech synthesis with next-distribution prediction," *arXiv preprint arXiv:2412.16846*, 2024.

[6] Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He, "Autoregressive image generation without vector quantization," *Advances in Neural Information Processing Systems*, vol. 37, pp. 56424–56445, 2024.

[7] William Peebles and Saining Xie, "Scalable diffusion models with transformers," *arXiv preprint arXiv:2212.09748*, 2022.

[8] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole, "Score-based generative modeling through stochastic differential equations," *arXiv preprint arXiv:2011.13456*, 2020.

[9] Zhihao Du, Qian Chen, Shiliang Zhang, Kai Hu, Heng Lu, Yexin Yang, Hangrui Hu, Siqi Zheng, Yue Gu, Ziyang Ma, et al., "Cosyvoice: A scalable multilingual zero-shot text-to-speech synthesizer based on supervised semantic tokens," *arXiv preprint arXiv:2407.05407*, 2024.

[10] Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, Jian Zhao, Kai Yu, and Xie Chen, "F5-tts: A fairytaler that fakes fluent and faithful speech with flow matching," *arXiv preprint arXiv:2410.06885*, 2024.

[11] Yuancheng Wang, Haoyue Zhan, Liwei Liu, Ruihong Zeng, Haotian Guo, Jiachen Zheng, Qiang Zhang, Shunsi Zhang, and Zhizheng Wu, "Maskgct: Zero-shot text-to-speech with masked generative codec transformer," *arXiv preprint arXiv:2409.00750*, 2024.

[12] Philip Anastassiou, Jiawei Chen, Jitong Chen, Yuanzhe Chen, Zhuo Chen, Ziyi Chen, Jian Cong, Lelai Deng, Chuang Ding, Lu Gao, Mingqing Gong, Peisong Huang, Qingqing Huang, Zhiying Huang, Yuanyuan Huo, Dongya Jia, Chumin Li, Feiya Li, Hui Li, Jiaxin Li, Xiaoyang Li, Xingxing Li, Lin Liu, Shouda Liu, Sichao Liu, Xudong Liu, Yuchen Liu, Zhengxi Liu, Lu Lu, Junjie Pan, Xin Wang, Yuping Wang, Yuxuan Wang, Zhen Wei, Jian Wu, Chao Yao, Yifeng Yang, Yuanhao Yi, Junteng Zhang, Qidi Zhang, Shuo Zhang, Wenjie Zhang, Yang Zhang, Zilin Zhao, Dejian Zhong, and Xiaobin Zhuang, "Seed-tts: A family of high-quality versatile speech generation models," *arXiv preprint arXiv:2406.02430*, 2024.

[13] Zhihao Du, Yuxuan Wang, Qian Chen, Xian Shi, Xiang Lv, Tianyu Zhao, Zhifu Gao, Yexin Yang, Changfeng Gao, Hui Wang, et al., "Cosyvoice 2: Scalable streaming speech synthesis with large language models," *arXiv preprint arXiv:2412.10117*, 2024.

[14] Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu, "Libritts: A corpus derived from librispeech for text-to-speech," *arXiv preprint arXiv:1904.02882*, 2019.

[15] Qwen team, "Qwen2 technical report," *arXiv preprint arXiv:2407.10671*, 2024.

[16] Matthew Le, Apoorv Vyas, Bowen Shi, Brian Karrer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, and Wei-Ning Hsu, "Voicebox: Text-guided multilingual universal speech generation at scale," 2023.

[17] Jonathan Ho and Tim Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598*, 2022.

[18] Jiaming Song, Chenlin Meng, and Stefano Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502*, 2020.

[19] Keyu An, Qian Chen, Chong Deng, Zhihao Du, Changfeng Gao, Zhifu Gao, Yue Gu, Ting He, Hangrui Hu, Kai Hu, Shengpeng Ji, Yabin Li, Zerui Li, Heng Lu, Haoneng Luo, Xiang Lv, Bin Ma, Ziyang Ma, Chongjia Ni, Changhe Song, Jiaqi Shi, Xian Shi, Hao Wang, Wen Wang, Yuxuan Wang, Zhangyu Xiao, Zhijie Yan, Yexin Yang, Bin Zhang, Qinglin Zhang, Shiliang Zhang, Nan Zhao, and Siqi Zheng, "Funaudiollm: Voice understanding and generation foundation models for natural interaction between humans and llms," *arXiv preprint arXiv:2407.04051*, 2024.

[20] Systran, "Faster whisper large v3," 2023.

[21] Zhifu Gao, Shiliang Zhang, Ian McLoughlin, and Zhijie Yan, "Paraformer: Fast and accurate parallel transformer for non-autoregressive end-to-end speech recognition," in *Interspeech*. 2022, pp. 2063–2067, ISCA.

[22] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong 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.

[23] Yafeng Chen, Siqi Zheng, Hui Wang, Luyao Cheng, Qian Chen, and Jiajun Qi, "An enhanced res2net with local and global feature fusion for speaker verification," *arXiv preprint arXiv:2305.12838*, 2023.

[24] Chun Yat Wu, Jiajun Deng, Guinan Li, Qiuqiang Kong, and Simon Lui, "Clear: Continuous latent autoregressive modeling for high-quality and low-latency speech synthesis," *arXiv preprint arXiv:2508.19098*, 2025.