Small-E: Small Language Model with Linear Attention for Efficient Speech Synthesis

Théodor Lemerle, Nicolas Obin, Axel Roebel

Small-E:
Small Language Model with Linear Attention for Efficient Speech Synthesis

Théodor Lemerle, Nicolas Obin, Axel Roebel

STMS Lab
IRCAM, CNRS, Sorbonne Université
Paris, France

Abstract

Recent advancements in text-to-speech (TTS) powered by language models have showcased remarkable capabilities in achieving naturalness and zero-shot voice cloning. Notably, the decoder-only transformer is the prominent architecture in this domain. However, transformers face challenges stemming from their quadratic complexity in sequence length, impeding training on lengthy sequences and resource-constrained hardware. Moreover they lack specific inductive bias with regards to the monotonic nature of TTS alignments. In response, we propose to replace transformers with emerging recurrent architectures and introduce specialized cross-attention mechanisms for reducing repeating and skipping issues. Consequently our architecture can be efficiently trained on long samples and achieve state-of-the-art zero-shot voice cloning against baselines of comparable size. Our implementation and demos are available at `https://github.com/theodorblackbird/lina-speech`.

Index Terms: speech synthesis, zero-shot adaptive text-to-speech, language modeling, linear attention

1. Introduction

1.1. Context and Related Works

Over the recent years, neural text-to-speech synthesis (TTS) has gained spectacular improvements in terms of quality with a diversity of approaches and paradigms [1, 2, 3, 4]. In particular, discrete speech and audio representations allowed immediate use of well-established decoder-only transformers such as GPT [5] in many state-of-the-art text-to-audio and text-to-speech model. However, transformers rely on the self-attention "time-mixing"[6] operation which can be efficiently trained in parallel but suffers from quadratic complexity with respect to the sequence length. The challenge of designing sequence modeling architecture that can compete with transformers has sparked a resurgence in research on recurrent neural networks (RNNs). This work introduces the broad term "linear attention" to denote this emerging class of RNNs that replaces self-attention for linear complexity "time-mixing" while keeping performances and high training throughput.

This paper primarily relates to speech models formulated as language models (LMs) or employing discrete audio codecs through Residual Vector Quantization (RVQ). VALL-E [7] employs an autoregressive transformer to predict the first quantizer and a parallel transformer for the residuals. Before the rise of RVQ codecs, Tortoise [8] achieved a significant improvement through scaling up and leveraged a decoder-only transformer to predict a VQ representation of the mel spectrogram. Some other works introduce semantic codes as low frame rate audio latents, following advancements in self-supervised speech representations. For instance Bark [9] separately predicts semantic codes from text, first quantizers from semantic codes, and residuals with three decoder-only transformers. SoundStorm [10] predicts audio from semantic codes in parallel by leveraging a MaskGit [11] architecture. In contrast, NaturalSpeech2 [12] avoids the language model formulation by learning the continuous latents of an RVQ codec with a diffusion model, sidestepping autoregressive modeling or semantic encoding and instead relying on given durations and fundamental frequency.

This research was supported by the project EXOVOICES ANR-21-CE23-0040 and funded by the French National Research Agency.

1.2. Linear surrogate of decoder transformer

Unlike previous RNNs such as LSTM or GRU, transformers are significantly faster to train, do not suffer from vanishing gradient and demonstrate scalability with parameters reaching into the hundreds of billions. Further hardware-aware implementation [13] of self-attention has established it as a prevalent choice for sequence modeling, including applications in audio processing. General softmax-based attention involves three sequences, denoted as $\mathbf{Q} \in \mathbb{R}^{N \times d}$, $\mathbf{K} \in \mathbb{R}^{N' \times d}$, and $\mathbf{V} \in \mathbb{R}^{N' \times d'}$, along with an optional mask $\mathbf{M} \in \mathbb{R}^{N \times N'}$. The attention function is defined as:

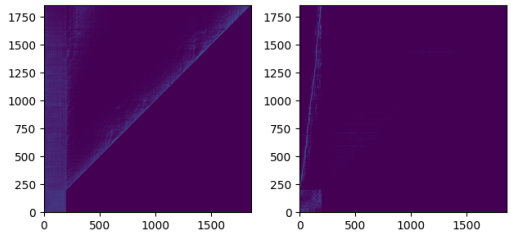
$$\text{Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}} \odot \mathbf{M})\mathbf{V}. \tag{1}$$

When $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ represent different linear projections of the same input sequence $\mathbf{X} \in \mathbf{R}^{N \times d}$ (and are therefore function of $\mathbf{X}$), the resulting function $\mathbf{X} \mapsto \text{Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$ is referred to as self-attention. Backing the success of GPT2 [5] and successors for natural language modeling, the decoder-only transformer architecture can be generalized with the terminology proposed in [6]:

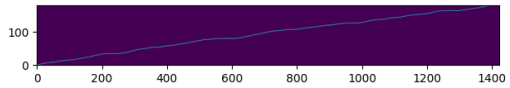
$$\begin{aligned} \mathbf{Y}' &= \mathbf{X} + \text{TimeMixing}(\text{Norm}(\mathbf{X})), \\ \mathbf{Y} &= \mathbf{Y}' + \text{ChannelMixing}(\text{Norm}(\mathbf{Y}')). \end{aligned} \tag{2}$$

Initially built with self-attention for TimeMixing [14], position-wise feed-forward network for ChannelMixing and layer normalization for Norm. In the context of autoregressive modeling, a causal mask $\mathbf{M}_{i,j} = \mathbf{1}_{i<j}$ is employed to prevent tokens from attending to relative future tokens, enabling the model to function as an autoregressive model while being trained in parallel.

However, self-attention exhibits quadratic complexity concerning sequence length during training and inference. Simultaneously, it has been observed that in certain scenarios, self-attention tends to focus on local reasoning [16], as evidenced in text-to-speech by almost diagonal attention weights in practice (see Figure 1). This observation suggests that computing every pair-wise relation in $\mathbf{Q}\mathbf{K}^T$ at every



(a) *In decoder-only LM TTS models [7, 9, 15], attention scores either boil down to cross-attention or local reasoning so that a large proportion of tokens are not attended.*



(b) *In our work, we only use two layers of cross-attention compared to self-attention in every layer.*

Figure 1: *Decoder-only attention weight tend to behave as an encoder-decoder.*

layer may not be essential. These insights align with recent developments in recurrent architectures which have emerged as potential replacements for transformers in natural language modeling. For instance, RWKV [6] is an RNN designed as an alternative to transformers. RWKV incorporates new mechanisms for both TimeMixing (known as "WKV" and is closely linked to some form of linear attention [17, 18, 19]) and ChannelMixing (involving linear interpolation of current and past token). In the lineage of State-Space Model [20], Mamba [21] unifies TimeMixing and ChannelMixing operations in (Equation 2), removes the linear time-invariant assumption with data-dependency and introduces parallelization through parallel scan. RetNet [22] features linear attention with decaying state for TimeMixing allowing efficient chunkwise computation. Gated Linear Attention (GLA) [23] explores hardware efficient chunkwise form of linear attention with data-dependent transition. All offer alternatives to the original transformer decoder-only block for language modeling with competitive throughput and performance for language modeling as demonstrated on various tasks. They scale linearly with the sequence length during training, opening the door to training on long sequences thus capturing long-term dependencies at a lower cost. We refer to them as **Linear Causal Language Model (LCLM)** blocks. To the best of our knowledge their usage for audio generative modeling remains largely unexplored.

1.3. TTS as conditional codec language modeling

The remarkable ability of language models to adapt from unseen inputs sample is known as "in-context learning" [24], and has been successfully adapted for TTS for zero-shot voice continuation [7]. For text-to-speech we follow conditional codec language modeling formulation as introduced by [7], given $\mathbf{x} = \{x_0, \dots, x_N\}$ a text transcription, $\mathbf{y} \in \{1, \dots, C\}^{Q \times T}$ a RVQ representation of the corresponding audio with Q quantizers of codebook size N_c we formulate it as

$$p(\mathbf{y}|\mathbf{x}) = \prod_{t=0}^{T} \prod_{q=1}^{Q} p(\mathbf{y_{q,t}}|\mathbf{x}, \mathbf{y_{<q,\leq t}}). \quad (3)$$

By concatenating source and target transcriptions and by providing only source audio tokens during inference, conditional codec modeling turns into a zero-shot voice cloning model without explicit need of speaker encoder module. In contrast with natural language modeling and because of the hierarchical nature of RVQ, we must account of the conditional dependencies between succeeding residuals. Previous work [7, 9] tend to train separate models for first "coarse" quantizers to "finer" successive residuals.

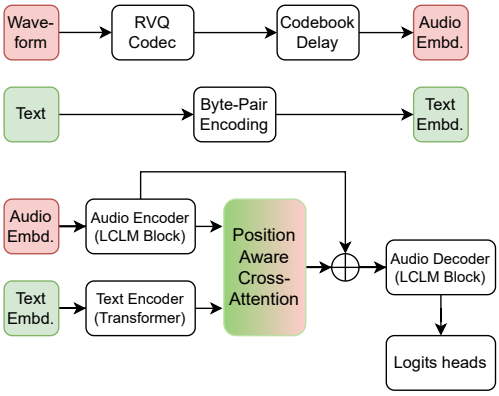


Figure 2: *Model Overview. Top: input pipeline. Bottom: encoder-decoder architecture.* $\bigoplus$ *means summation.*

1.4. Positioning and contributions

Being able to train on large datasets is a crucial aspect for diverse and expressive speech generation. Typical LM are difficult to train in the limited hardware regime (*ie* consumer grade GPU). Our model shows that language modeling can be successfully adapted to the small model regime with careful architecture considerations :

- This paper introduces **Linear Causal Language Model (LCLM)** blocks instead of autoregressive transformers commonly used in language modeling for audio application. To the best of our knowledge this is the first time they are used for text-to-speech. As a consequence we are able to train efficiently on long samples (up to 30s). We hypothesize that it is crucial for learning expressive speech.
- We introduce a **Position-Aware Cross-Attention (PACA)** mechanism which is specifically designed for text-to-speech and helps with skipping and repeating issues.
- The proposed model has competitive performance on zero-shot voice cloning TTS by comparison to existing TTS models of the same size, while requiring much less resources during training.

2. Small-E

This section presents Small-E, a multi-speaker neural TTS with zero-shot voice cloning capabilities. Small-E belong to the family of neural codec language model such as [7, 9, 15, 10]. In contrast with previous TTS codec LM model that leverages decoder-only (GPT) transformers, Small-E relies on encoder-decoder architecture. Indeed, we observed that previous decoder-only transformers tend to behave internally as an encoder-decoder (see Figure 1) leading to a potential waste of compute. The general architecture is presented in Figure 2, text is encoded through a non-causal transformer, audio is encoded with a stack of LCLM blocks. Both encoders outputs are fed to the cross-attention that learns to align text to audio. The audio decoder (same as the audio encoder) takes audio embeddings and cross-attention output. The decoder output is projected to logits.

2.1. Model architecture

The input pipeline for audio and text compression is processed in the following manner. Audio is compressed with an RVQ codec. We employ a codebook delaying scheme introduced by

MusicGen [25] in order to enforce the conditional dependencies between residual codebooks. Text is compressed with byte-pair encoding before embedding. Then the text embedding is processed with an non-causal transformer encoder, the audio embeddings are processed with an encoder consisting of a stack of LCLM blocks. Both encoder outputs are then fed to a Position-Aware Cross-Attention, the output being the text embedding attended for each audio embedding. The text embedding and audio embedding are then superposed and fed to an audio decoder similar to the audio encoder. The three basic blocks are:

Text Encoder This component comprises a stack of non-causal (parallel) transformer encoders with RoPE positional embedding [26].

Audio Encoder/Decoder We investigated various LCLM blocks including RWKV [6] v5.2/v6, Mamba [21] and GLA [23]. In early experiments we found that they offer comparable performances.

Position-Aware Cross-Attention (PACA) Autoregressive speech modeling is prone to skipping and repeating issues [7, 27, 28]. We introduce a simple tweak to enforce position awareness. In the conventional formulation of cross-attention between text and audio, represented as:

$$\mathbf{Y} = \text{Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V}), \tag{4}$$

where $\mathbf{Q}$ is the audio latent sequence, and $\mathbf{K}$, $\mathbf{V}$ are linear projections of the text latent sequence, attention is computed independently for every time step, without considering previous attended text latent.

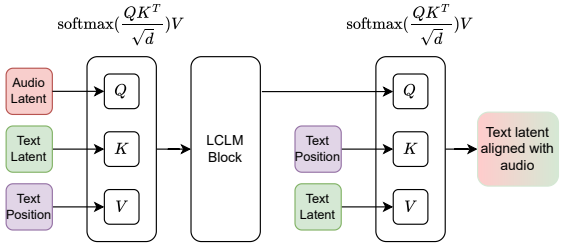


Figure 3: *Position-Aware Cross-Attention*

To address this limitation, we propose a modification to the cross-attention mechanism by explicitly materializing position information along with a feedback loop to propagate past positions (see Figure 3). Firstly, a cross-attention is computed by selecting text positions only:

$$\mathbf{Y^{(1)}} = \text{Att}(\mathbf{Q}, \mathbf{K}, \mathbf{P}), \tag{5}$$

where $\mathbf{P}$ is sinusoidal positional embedding along the text latent positions, that is:

$$\begin{aligned} \mathbf{P_{t,2d}} &= \sin(t/10000^{2d/d_b}), \\ \mathbf{P_{t,2d+1}} &= \cos(t/10000^{2d/d_b}). \end{aligned} \tag{6}$$

This ensures that $\mathbf{Y^{(1)}}$ contains only information about the position of the attended text latent, rather than actual text content. These positions are then fed into a LCLM block represented by the transition function f to introduce a feedback loop on the positions:

$$\mathbf{Y^{(2)}_{t+1}}, \mathbf{H_{t+1}} = \mathbf{f}(\mathbf{Y^{(1)}_t}, \mathbf{H_t}). \tag{7}$$

This causal linear LCLM block can be any recurrent LM block such as RWKV [6], Mamba [21] or GLA [23].

Finally, a cross-attention of $\mathbf{Y_2}$ against $\mathbf{P}$ is performed to select $\mathbf{V}$ (containing text information), mapping the position $\mathbf{Y^{(2)}}$ to the text latent:

$$\mathbf{Y^{(3)}} = \text{Att}(\mathbf{Y^{(2)}}, \mathbf{P}, \mathbf{V}). \tag{8}$$

It is important to note that the positional embedding $\mathbf{P}$ is not superposed onto any latent vector but rather materialized independently, constraining the model to accurately encode positional information to effectively attend to the text content. We set d_b to be significantly smaller than the model dimension ($d_b \leq 64$) to keep additional operations negligible. This approach is reminiscent of Location Sensitive Attention [27], while being an order of magnitude faster due to the efficiency of LCLM blocks. The optimization objective is cross-entropy loss between original RVQ codec and logits prediction.

3. Experimental Evaluation

3.1. Dataset

Small-E was trained on Librilight medium [29], consisting of approximately 5,000 hours of multi-speaker English speech recordings reading audio books, collected from LibriVox. We used the provided recipes to get samples of approximately 25 seconds. Speech utterances were transcribed textually using Ocotillo [30] speech recognition system. Validation set is made of 2,000 random utterances. For evaluation, the proposed model and the benchmark models (see below) were compared on the LibriTTS test split [29], in particular by insuring training and testing speakers do not overlap.

3.2. Implementation details

For the text encoding, the proposed model used byte-pair encoding with a vocabulary size of 256 computed on the dataset transcription. For the audio encoding we used EnCodec [31] at 3kbps bitrate. In a preliminary experiment, the proposed model has been compared with different LCLM blocks, including RWKV, Mamba, and GLA. For each, the text encoder consists of 9 layers of non-causal transformer, each layer of dimension 512 with 8 heads. For the experiments, we chose Gated Linear Attention as LCLM Block, each block consisting of 6 layers of inner dimension 512 with 2 heads. We observed during this preliminary that GLA is performing similarly as Mamba and RWKV in terms of validation loss while giving slightly better training throughput with respect to our configuration (i.e fixed batch size and number of parameters). For this reason, the proposed model is using GLA blocks in the remaining of this paper. The whole model consists of 64M trainable parameters. For the training, Adam optimizer was used with a learning rate equal to $5\text{e}{-4}$, with momentum $\beta_1 = 0.9$, $\beta_2 = 0.999$, and weight decay of 0.1. We group sentences of similar length within 10 buckets and use dynamic batch size with target size of approximately $80,000$ audio tokens. We used gradient clipping of 1.0. Trainings were done on 4 RTX3080 (10GB VRAM each) during two days, consisting of 15 epochs over the dataset. During inference we use top-k sampling with k set to 100 for the first quantizer and greedy decoding for the residuals.

3.3. Benchmark

We compared Small-E with YourTTS [32] which is a common baseline for the evaluation of multi-speaker TTS models (e.g., [7, 12]). For the comparison, we used the official checkpoint which has been trained on a multilingual dataset comprising VCTK (English), LibriTTS (English), and Portuguese split of MLS. In addition, we also compared with MetaVoice [15] as a strong baseline, an open-source and open-weight GPT model of 1.2B parameters trained on 100k hours of speech from a private dataset. This constitutes to our knowledge the strongest

codec language model TTS model publicly available. Notably for decoding EnCodec tokens, our model rely on Vocos [33] at 3kbps. This is in contrast with MetaVoice which leverages MultiBand diffusion at 6kbs and an additional post-net. We regret that most of the baselines belonging to our family don't have publicly available official implementations, limiting our subjective and objective evaluation.

3.4. Methodology

3.4.1. Objective evaluation

As for the quantitative objective evaluation, we investigated the performance of the proposed Small-E architecture in terms of training throughput, i.e., the throughput measured by means of audio-tokens per seconds and the perplexity of the LM as an indicator of the reconstruction error as the exponential of the cross-entropy loss. This is measured on Librilight medium with the setup described in Section 3.2. Additionally, we conducted an ablation study to investigate the role of the proposed PACA mechanism with respect to the skips and repetitions problem known as a common issue of auto-regressive models [7, 27, 34]. To do so, we followed the methodology presented in [34]. 100 utterances were randomly picked up from the validation set and were manually inspected in terms of skips and repetitions by comparison of the reference utterance. This methodology is preferred to the common measurement of the word error rate since the skip and repetition problem is specific to auto-regressive models [34, 7].

3.4.2. Subjective Evaluation

A subjective evaluation was additionally conducted to assess the *naturalness* and the *similarity* to the reference speaker of the considered speech sample. The experiment consisted in presenting to the participants a speech sample and a reference speech sample of the same speaker but pronouncing another utterance. The participants were asked to judge the speech sample with respect of the following instructions on a 5-degree MOS scale : (1) *naturalness*: *to which extent the speech sample is judged as natural as real human speech?*; (2) *similarity* : to the reference speaker: *to which extent the speech sample is judged close to the reference speaker?* For each participant subject, an experiment run consisted into the judgement of 15 samples. These samples were randomly selected among a total of 50 utterances (the same for all models) $\times$ 4 models (the three models being and the real speech = 200 speech samples. The real speech was presented as a positive anchor to the participant. The whole experiment has been conducted using the Prolific platform with a mix of 70 native and non-native English speakers.

4. Results and Discussion

Table 1 presents the training throughput of Small-E with comparison to a standard decoder-only architecture (following [7] implementation) taken as a baseline of the LMs generative family. Small-E training is significantly faster compared to this baseline architecture with same amount of parameters, with a relative increase of 62 %. This comes with a slight improvement of the perplexity, indicating that training throughput gain doesn't come at the cost of performance.

Table 2 presents the results of the ablation study. On the 100 generated utterances which were manually inspected, the Small-E version with PACA presents a drastic reduction of this problem, either in terms of skips or repetitions. This proves the efficiency of the proposed cross-attention mechanism as a solution to the skip and repetition problem.

[1] Using EnCodec[31] at 3kbps with delaying scheme[25], it consists of 4 tokens generated in parallel per decoding step.

Table 1: *Training throughput. Throughput is measured by means of audio token per second (in kilo tokens per second), and perplexity (referred to as ppl).*

Model	Audio token per second (kT/s) [1] ↑	ppl ↓
Small-E	316	18.33
Decoder-only (GPT)	195	19.68

Table 2: *Position-Aware Cross-Attention impact on 100 utterances. Number of utterances that contains at least one skip/repetition.*

Model	Skip ↓	Repeat ↓
Small-E w. PACA	1	1
w/o PACA	5	9

Table 3 presents the MOS scores obtained for the subjective evaluation. Under t-test, we found every pair of candidates to be significantly different ($p < 0.05$). Firstly, we observe that Vocos at low bitrate presents a slight but significant degradation with comparison to the original speech sample. Secondly, Small-E presents significantly higher scores than the baseline YourTTS both in terms of naturalness and similarity. Finally, Small-E presents significantly lower score than the strong baseline MetaVoice as expected since it consists of 20 times more data and parameters, excluding it from training on limited hardware.

Table 3: *Subjective evaluation. MOS for naturalness, and SMOS for similarity to the reference speaker.*

Model	Params.	MOS ↑	SMOS ↑
Original		4.55 ± 0.20	4.62 ± 0.23
Vocos 3kbps		4.27 ± 0.21	4.43 ± 0.22
Small-E (ours)	64M	3.16 ± 0.28	3.08 ± 0.30
YourTTS	86M	2.56 ± 0.24	2.54 ± 0.24
MetaVoice	1.2B	3.80 ± 0.28	3.91 ± 0.28

5. Conclusion

In this paper we presented Small-E, a TTS model based on codec language model. The proposed model tackles limitations of current LM TTS models. Firstly, we introduced Linear Causal Language Model in place of the traditional decoder-only transformer. Secondly, we introduced a cross-attention mechanism designed specifically to handle text and speech modalities in the context of TTS, with the idea of preventing the skip and repetition problem of auto-regressive models. In contrast with existing work, we were able to show that training LM TTS model is interesting even on limited hardware and leads to state-of-the-art quality against model of the same size. Experimental evaluation demonstrated the efficiency of the proposed model either in terms of training throughput, skip and repetition reduction, as well as naturalness and similarity to the reference speaker of the generated speech. These observations constitute encouraging results opening the way for small and efficient generative TTS models. In future work we are interested in streaming TTS, taking advantage of the linear complexity of LCLM for very long or embedded synthesis.

6. References

[1] 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," *International Conference on Learning Representations (ICLR)*, 2021.

[2] 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 (ICML)*, 2021, pp. 5530–5540.

[3] M. Le, A. Vyas, B. Shi, B. Karrer, L. Sari, R. Moritz, M. Williamson, V. Manohar, Y. Adi, J. Mahadeokar *et al.*, "Voicebox: Text-guided Multilingual Universal Speech Generation at Scale," *Conference in Neural Information Processing Systems (NeurIPS)*, 2023.

[4] Y. Guo, C. Du, Z. Ma, X. Chen, and K. Yu, "Voiceflow: Efficient text-to-speech with rectified flow matching," *arXiv preprint arXiv:2309.05027*, 2023.

[5] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language Models are Unsupervised Multitask Learners," 2019.

[6] B. Peng, E. Alcaide, Q. G. Anthony, A. Albalak, S. Arcadinho, S. Biderman, H. Cao, X. Cheng, M. Chung, M. Grella, G. Kranthikiran, X. He, H. Hou, P. Kazienko, J. Kocoń, J. Kong, B. Koptyra, H. Lau, K. S. I. Mantri, F. Mom, A. Saito, X. Tang, B. Wang, J. S. Wind, S. Wozniak, R. Zhang, Z. Zhang, Q. Zhao, P. Zhou, J. Zhu, and R. Zhu, "RWKV: Reinventing RNNs for the Transformer Era," in *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

[7] 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.

[8] J. Betker, "Better Speech Synthesis through Scaling," *arXiv preprint arXiv:2305.07243*, 2023.

[9] Suno, "Bark," https://github.com/suno-ai/bark, 2023.

[10] Z. Borsos, M. Sharifi, D. Vincent, E. Kharitonov, N. Zeghidour, and M. Tagliasacchi, "SoundStorm: Efficient Parallel Audio Generation," *arXiv preprint arXiv:2305.09636*, 2023.

[11] H. Chang, H. Zhang, L. Jiang, C. Liu, and W. T. Freeman, "MaskGIT: Masked Generative Image Transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 11 315–11 325.

[12] 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," in *Submitted to International Conference on Learning Representations (ICLR)*, 2024.

[13] T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré, "FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2022, pp. 16 344–16 359.

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is All you Need," in *Advances in Neural Information Processing Systems (NIPS)*, vol. 30, 2017.

[15] MetaVoice, "Metavoice-1b," https://github.com/metavoiceio/metavoice-src, 2024.

[16] T. Parcollet, R. van Dalen, S. Zhang, and S. Bhattacharya, "SummaryMixing: A Linear-Complexity Alternative to Self-Attention for Speech Recognition," in *submitted to Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

[17] N. Kitaev, L. Kaiser, and A. Levskaya, "Reformer: The Efficient Transformer," in *International Conference on Learning Representations (ICLR)*, 2020.

[18] Z. Shen, M. Zhang, H. Zhao, S. Yi, and H. Li, "Efficient attention: Attention with linear complexities," in *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2021, pp. 3531–3539.

[19] S. Zhai, W. Talbott, N. Srivastava, C. Huang, H. Goh, R. Zhang, and J. Susskind, "An Attention Free Transformer," in *International Conference on Learning Representations (ICLR)*, 2021.

[20] A. Gu, K. Goel, and C. Ré, "Efficiently Modeling Long Sequences with Structured State Spaces," in *International Conference on Learning Representations (ICLR)*, 2022.

[21] A. Gu and T. Dao, "Mamba: Linear-Time Sequence Modeling with Selective State Spaces," in *Submitted to International Conference on Learning Representations (ICLR)*, 2024.

[22] Y. Sun, L. Dong, S. Huang, S. Ma, Y. Xia, J. Xue, J. Wang, and F. Wei, "Retentive Network: A Successor to Transformer for Large Language Models," in *submitted to International Conference on Learning Representations (ICLR)*, 2023.

[23] S. Yang, B. Wang, Y. Shen, R. Panda, and Y. Kim, "Gated Linear Attention Transformers with Hardware-Efficient Training," *arXiv preprint arXiv:2312.06635*, 2023.

[24] I. Lee, N. Jiang, and T. Berg-Kirkpatrick, "Exploring the Relationship Between Model Architecture and In-Context Learning Ability," *arXiv preprint arXiv:2310.08049*, 2023.

[25] J. Copet, F. Kreuk, I. Gat, T. Remez, D. Kant, G. Synnaeve, Y. Adi, and A. Défossez, "Simple and Controllable Music Generation," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[26] J. Su, Y. Lu, S. Pan, A. Murtadha, B. Wen, and Y. Liu, "RoFormer: Enhanced Transformer with Rotary Position Embedding," *Neurocomputing*, vol. 568, p. 127063, 2024.

[27] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan, R. A. Saurous, Y. Agiomvrgiannakis, and Y. Wu, "Natural TTS Synthesis by Conditioning Wavenet on Mel Spectrogram Predictions," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4779–4783.

[28] R. Badlani, A. Łańcucki, K. J. Shih, R. Valle, W. Ping, and B. Catanzaro, "One TTS Alignment to Rule Them All," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 6092–6096.

[29] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P. E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux, "Libri-Light: A Benchmark for ASR with Limited or No Supervision," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7669–7673.

[30] J. Betker, "ocotillo - a fast, accurate and super simple speech recognition model," https://github.com/neonbjb/ocotillo, 2022.

[31] A. Défossez, J. Copet, G. Synnaeve, and Y. Adi, "High Fidelity Neural Audio Compression," *arXiv preprint arXiv:2210.13438*, 2022.

[32] E. Casanova, J. Weber, C. Shulby, A. C. Junior, E. Gölge, and M. A. Ponti, "YourTTS: Towards Zero-Shot Multi-Speaker TTS and Zero-Shot Voice Conversion for everyone," 2023.

[33] H. Siuzdak, "Vocos: Closing the Gap between Time-Domain and Fourier-based Neural Vocoders for High-Quality Audio Synthesis," in *International Conference on Learning Representations (ICLR)*, 2023.

[34] E. Georgiou, K. Kritsis, G. Paraskevopoulos, A. Katsamanis, V. Katsouros, and A. Potamianos, "Regotron: Regularizing the Tacotron2 Architecture Via Monotonic Alignment Loss," in *IEEE Spoken Language Technology Workshop (SLT)*, 2023, pp. 977–983.