Seed-Music: A Unified Framework for High Quality and Controlled Music Generation

Seed Team, ByteDance*

Abstract

We introduce Seed-Music, a suite of music generation systems capable of producing high-quality music with fine-grained style control. Our unified framework leverages both auto-regressive language modeling and diffusion approaches to support two key music creation workflows: *controlled music generation* and *post-production editing*. For controlled music generation, our system enables vocal music generation with performance controls from multi-modal inputs, including style descriptions, audio references, musical scores, and voice prompts. For post-production editing, it offers interactive tools for editing lyrics and vocal melodies directly in the generated audio.

We encourage readers to listen to demo audio examples at https://team.doubao.com/seed-music.

1 Introduction

Music is deeply embedded in human culture. Throughout human history, vocal music has accompanied key moments in life and society: from love calls to seasonal harvests [Mehr et al., 2019]. Today, vocal music remains central to global culture. However, creating vocal music is a complex, multi-stage process involving pre-production, writing, recording, editing, mixing, and mastering [Owsinski, 2010, Senior, 2012], making it challenging for most people. Our goal is to leverage modern generative modeling technologies, not to replace human creativity, but to lower the barriers to music creation. By offering interactive creation and editing tools, we aim to empower both novices and professionals to engage at different stages of the music production process.

Today, deep generative models are capable of understanding and generating multi-modal data [Yin et al., 2023, Zhan et al., 2024]. Although music generation has benefited from the advances in natural language processing [Radford et al., 2019], speech synthesis [Anastassiou et al., 2024], and computer vision [Peebles and Xie, 2023], the task remains challenging for several reasons:

- **Domain Complexity.** Music signals are highly complex, exhibiting both short-term melodic coherence and long-term structural consistency. Unlike speech, vocal music features overlapping sounds across a wide frequency range. Singing, with its wide pitch range and expressive techniques, adds another layer of complexity. The model must simultaneously generate melodic vocals, harmonic tones, and rhythmic percussion.
- **Evaluation Difficulty.** Evaluating music generation models often requires domain expertise to assess artistic quality. This includes judging the appeal of melodies, the coherence of chord progressions, the presence of idiomatic structure, and the expressiveness of vocals. Many of these aesthetic qualities are deeply influenced by cultural and regional differences. Quantifying these artistic elements in music poses a challenge.
- **Data Complexity.** For generative models to learn how to condition outputs on lyrics, genre, instrumentation, and song structure, annotated music data is essential. Music annotation requires

*Please cite this work as "Seed-Music (2024)". The full statement of author contributions and acknowledgments can be found at the end of the document. Correspondence regarding this technical report should be sent to Seed-Music@bytedance.com.

special domain knowledge. While many people can transcribe speech or label images, identifying musical elements such as chords, song sections, instruments, and genres requires a strong musical background.

- **Diverse User Segments and Needs.** The needs of novice musicians differ greatly from those of professionals. A text-to-music system [Agostinelli et al., 2023, Copet et al., 2024] that generates a complete audio piece from a language prompt can be transformative for a beginner, but may offer limited value to professional producers, who typically seek more granular control over compositions and access to individual instrument stems. Even among professionals, needs differ: a guitarist might need vocal editing tools, while a vocalist might want to tweak guitar or other instrument tracks.

Contributions With these challenges in mind, we highlight the versatility of Seed-Music. It supports vocal and instrumental music generation, singing voice synthesis, singing voice conversion, music editing and more. Our methodology, experiments, and solutions are designed to address diverse use cases. Rather than relying on a single modeling approach, such as auto-regression (AR) or diffusion, we propose a unified framework that adapts to the evolving workflows of musicians. Our key contributions are threefold:

- We introduce an auto-regressive language model (LM) based approach for high-quality vocal music generation conditioned on diverse and multi-modal user inputs.
- We present a diffusion based approach for fine-grained music audio editing.
- We propose a novel method for zero-shot singing voice conversion, which requires only 10-second singing or speech recording from a user.

In Section 2, we give a brief literature review of music generation. In Section 3, we introduce our unified framework consisting of three parts: 1) a representation learning module that converts raw audio into a highly compressed representation, 2) a generation module that predicts the representation based on various user controls, and 3) a rendering module trained to produce the waveform from the representation. Our framework establishes three representations for different scenarios: audio tokens, symbolic tokens, and vocoder latents. The corresponding design choices will be detailed. In Section 4, we dive into how our unified framework can be configured and trained to support various music generation and editing tasks. In Section 5 and Section 6, we discuss potential applications and limitations of Seed-Music, including those related to building safe and ethical generative AI systems.

2 Literature Review

Vocal music can be defined as the combination of a vocal track and an instrumental track. Historically, full mixes of vocal music were created by generating these two tracks separately and mixing them together. With advances in ML-based generative modeling, Jukebox [Dhariwal et al., 2020] was one of the first published works to output full mixes of vocal music in an end-to-end fashion. Although prohibitively slow, Jukebox could generate vocal music that closely reflected input lyrics and specified artist and genre. Fast forward to today, many AI music creation apps enable creators to instantly generate vocal music on-demand by providing natural language prompts. The field of music generation has a long history. Music generation systems can largely be divided into two categories: symbolic domain generation and audio domain generation. A few representative works are covered in this section.

Symbolic music based systems The earliest instrumental music systems were rule-based and generated notes in limited styles [Cope, 1989]. These systems generate symbolic representation, which is then rendered into audio using digital signal processing (DSP) methods. Later, rule-based approaches were replaced with data-driven approaches for music generation. Examples include FolkRNN [Sturm et al., 2016], PerformanceRNN [Simon and Oore, 2017], MusicTransformer [Huang et al., 2018], MuseNet [OpenAI], CoCoNet [Huang et al., 2019] and many others. Although many systems relied mainly on MIDI (Music Instrument Digital Inferfaces) data, other symbolic notation such as ABC notation, tablature and music XML [Dong et al., 2020, Sarmento et al., 2021] were also explored. The defining limitation for all these systems was the limited availability of labelled training data. MIDI transcriptions are notoriously difficult, time-consuming and expensive to procure.

Audio renderers for symbolic music based systems A symbolic system, whether rule-based or data-driven, requires an audio renderer to produce sound. A traditional audio renderer employs DSP-based synthesizers and samplers to render instrument sounds. A parallel line of research applied a data-driven approach to audio synthesis, which is often referred as "neural audio synthesis". Examples include NSynth [Engel et al., 2017], GANSynth [Engel et al., 2019], DDSP [Engel et al., 2020], MIDI-DDSP [Wu et al., 2022], WAVERNN-based approaches [Kalchbrenner et al., 2018, Hantrakul et al., 2019]. These techniques opened up new timbral and tonal possibilities compared to traditional DSP-based audio synthesis [Serra and Smith, 1990]. For example, a WaveNet-based audio render can generate expressive and nuanced piano recordings from generated piano scores [Hawthorne et al., 2019]. In the vocal generation domain, a Singing Voice Synthesis (SVS) can render vocal performances based on notes and phonemes generated by a symbolic system [Cook, 1996, Nishimura et al., 2016, Yi et al., 2019, Lu et al., 2020, Zhuang et al., 2021].

Language model generative models Breakthroughs in language modeling gave birth to a new way of generating musical audio. AudioLM [Borsos et al., 2023] demonstrated waveform generation could be formulated as a next-token prediction task where tokens are extracted from a neural codec such as SoundStream [Zeghidour et al., 2021]. In this light, we are inspired by LM based systems in speech synthesis, including a related family of Seed Models [Anastassiou et al., 2024, Bai et al., 2024], the family of VALLE-style models [Wang et al., 2023a, Zhang et al., 2023, Chen et al., 2024a] and many others [Betker, 2023, Łajszczak et al., 2024].

In music generation, LM based modeling marks a paradigm shift. MusicLM [Agostinelli et al., 2023] could train directly on full mixes of music at unprecedented scales. These new end-to-end systems are composed of a hierarchy of modules that blur the historical separation of symbolic generation and audio synthesis. From a high level, the singing content and the music style are captured by "semantic tokens", which are manipulated by LMs at earlier stages of the hierarchy, while instrumental timbre and high frequency details are captured by "acoustic tokens" and manipulated in later stages of the hierarchy.

Additionally, an LM-based approach enables other modalities like text to be effectively transformed into music. Instead of procuring an impossible amount of paired text-music data, MusicLM circumvented this need by extracting semantic tokens from pre-trained models like MuLan [Huang et al., 2022] and Wav2Vec-BERT [Chung et al., 2021]. An LM-based model then transforms these semantic tokens into target audio tokens representing music. Follow-up works usually have a similar high level recipe: a neural audio codec tokenizes the audio into discrete codes, a GPT2-like system [Vaswani et al., 2017, Radford et al., 2019, Touvron et al., 2023] performs next-token prediction of audio tokens based on multi-modal inputs that are also casted into token format, and the neural audio codec futher renders the waveform based on the predicted token sequence. These methods can also be applied to singing vocals [Li et al., 2024a,b, Huang et al., 2024].

Diffusion-based generative models An explosion of diffusion-based systems are revolutionizing the way images, video and audio are generated. We take inspiration from the vision field ([Ho et al., 2020, Song et al., 2022, 2021, Rombach et al., 2022]) and adapt these for music generation.

For instrumental music generation, a variety of works successfully implemented text-to-music based on a diffusion backbone: Noise2Music [Huang et al., 2023a], Stable Audio [Evans et al., 2024a], Stable Audio Open [Evans et al., 2024b], MUSTANGO [Melechovsky et al., 2024] and MusicLDM [Chen et al., 2023, Liu et al., 2024]. From a high level, instead of modeling high-dimensional waveforms directly in the diffusion process, these recipes often employ latent diffusion [Rombach et al., 2022] on normalized latents extracted from a vocoder that was trained separately. The process of denoising the intermediate representation can then be conditioned on various music annotations to support fine-grain and controlled generation. These include contour-based and downbeat conditioning in Music-ControlNet [Wu et al., 2023], chroma-based melody in Music-Gen [Copet et al., 2024] or mixtures of the above in JASCO [Tal et al., 2024].

In the vision domain, diffusion-based systems have generalized to tasks beyond generation from scratch. These include in-painting, style transfer and contour-based editing [Meng et al., 2021, Couairon et al., 2022, Su et al., 2023, Kawar et al., 2023, Esser et al., 2024a]. We note many exciting analogues in the vocal music domain. For example, recent singing voice synthesis (SVS) [Liu et al., 2022] and singing voice conversion (SVC) [Li et al., 2024c, Chen et al., 2024b, Yamamoto et al., 2023] approaches can be thought of as a type of controllable style transfer.

Representation Learning In both LM-based and diffusion-based generative modeling, a critical component is the choice of representation for the audio signal. Methods for encoding speech, environmental sound and music exist on a spectrum ranging from semantic representations to acoustic representations. On one end, CLAP [Elizalde et al., 2022] and MuLan [Huang et al., 2022] are joint music-text representations that largely capture the semantic content of the underlying music and language description. On the other end, auto-encoder style models like WaveVAE [Peng et al., 2020] or Music2Latent [Pasini et al., 2024] compress waveform into an n-dimensional continuous latent space that largely captures acoustic content. Discrete neural audio codecs such as SoundStream [Zeghidour et al., 2021], Encodec [Défossez et al., 2022], Descript Audio Codec [Kumar et al., 2023], HiFi-Codec [Yang et al., 2023] and WavTokenizer [Ji et al., 2024] capture acoustic content as a series of quantized codebooks of increasing frequency detail. Spectrogram-derived approaches such as AudioMAE [Huang et al., 2023b] and Wav2Vec-BERT [Chung et al., 2021] are pretraining stages capturing varying mixtures of acoustic and semantic information suited to the downstream task. Some approaches operate on spectrograms directly as part of a text-based in-painting framework like VoiceBox [Le et al., 2024] or AudioBox [Vyas et al., 2023] whilst others may even use aggressively down-sampled waveform in a diffusion-based framework [Huang et al., 2023a] or VQ-VAE framework [Dhariwal et al., 2020]. Depending on the task at hand, these approaches present varying levels of advantages and trade-offs. What type of representation is most suitable for music, methods to compress short term and long term features, mechanisms for encoding melodic and rhythmic domain knowledge and the trade-off between semantic and acoustic information continues to be active areas of work.

3 Method

Our music generation system consists of three core components as illustrated in Figure 1: a **Representation Learning module**, which compresses the raw audio waveform into the intermediate representation that serves as the foundation for training the subsequent components; a **Generator**, which processes various user control inputs and generates the corresponding intermediate representation; and a **Renderer**, which synthesizes high-quality audio waveform based on the intermediate representation from Generator.

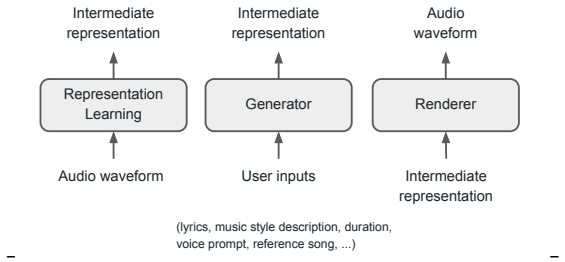

Figure 1: An overview of Seed-Music framework.

The primary design choice is the intermediate representation. As outlined in section 2, we identify three practical options: audio tokens, symbolic music tokens, and vocoder latents. The advantages and limitations of each are summarized in Table 1.

Representation	Compression	Interpretability	Renderer friendly	Generator friendly
Audio token	yes	no	maybe	yes
Symbolic music	yes	yes	no	yes
Vocoder latent	maybe	no	yes	maybe

Table 1: Comparison of intermediate representations.

- **Audio tokens** are designed to efficiently encode both semantic and acoustic information at a much lower token rate than the audio sampling rate [Agostinelli et al., 2023, Copet et al., 2024, Dhariwal et al., 2020]. When used with an auto-regressive LM based Generator, audio tokens

serve as effective representations for connecting different modalities. However, their primary limitation lies in their lack of interpretability. Musical attributes such as vocal pronunciation, timbre, and pitch are embedded in a highly entangled format. Previous work [Borsos et al., 2023] has explored how some audio tokens correspond to semantic features, while others capture acoustic aspects. This entanglement makes it challenging for the Generator to control specific elements of music, like melody and timbre, during audio token generation.

- **Symbolic representations**, such as MIDI, ABC notation and MusicXML, are discrete and can be easily tokenized into a format compatible with LMs. Unlike audio tokens, symbolic representations are interpretable, allowing users to read and modify them directly. However, their lack of acoustic details means the system has to rely heavily on the Renderer's ability to generate nuanced acoustic characteristics for musical performance. Training such a Renderer requires large-scale datasets of paired audio and symbolic transcriptions, which are especially scarce for vocal music.
- **Vocoder latents** from a variational auto-encoder serve as continuous intermediate representations, especially when used with diffusion models [Evans et al., 2024b]. These latents capture more nuanced information compared to quantized audio tokens, allowing for a lighter Renderer in this pipeline. However, similar to audio tokens, vocoder latents are uninterpretable. Moreover, since vocoder latents are optimized for audio reconstruction, they may encode too much acoustic detail that is less useful for the Generator's prediction tasks.

The selection of an intermediate representation depends on the specific downstream music generation and editing tasks. In the rest of this section, we present the technical details of our system design with these three intermediate representations and showcase their applications in Section 4.

3.1 Audio Token-based Pipeline

The audio token-based pipeline, as illustrated in Figure 2, includes four building blocks: (1) an audio tokenizer, which converts raw music waveforms into low rate discrete tokens; (2) an auto-regressive LM (i.e. the Generator), which takes in user control inputs, convert them into prefix tokens, and predicts a sequence of target audio tokens; (3) a token diffusion model, which predicts the vocoder latents based on the audio tokens; and (4) an acoustic vocoder, which renders the final 44.1kHz stereo audio waveform. The token-to-latent diffusion module and latent-to-waveform vocoder module collectively form the token-to-waveform process, referred to as the Renderer.

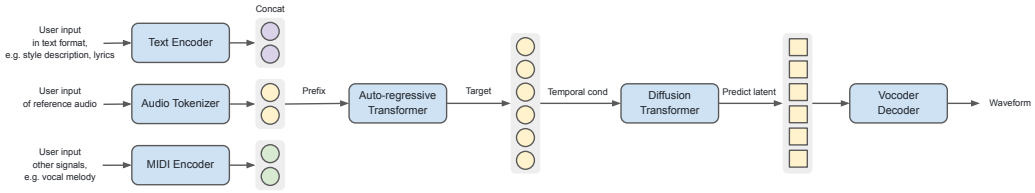

Figure 2. Overview of the Seed-Music pipeline with audio token as intermediate representation. (1) Input embedders convert multi-modal controlling inputs, such as music style description, lyrics, reference audio, or music scores, into a prefix embedding sequence. (2) The auto-regressive LM generates a sequence of audio tokens. (3) The diffusion transformer model generates continuous vocoder latents. (4) The acoustic vocoder produces high-quality 44.1kHz stereo audio.

Audio tokenizer The effectiveness of the audio tokenizer is critical to the success of this pipeline. The audio tokens embed key musical information from the original signals, such as melody, rhythm, harmony, phonemes, and instrument timbre. Our implementation is inspired by Betker [2023], Wang et al. [2023b], and Łajszczak et al. [2024], with further optimizations in architecture and training to achieve the following:

- High retention of essential information at a low compression rate, improving the training efficiency of the auto-regressive LM.
- A balance between semantic and acoustic features, ensuring sufficient semantic details to optimize the Generator training while maintaining enough acoustic details for accurate waveform reconstruction by the Renderer. This trade-off between token generation and signal reconstruction [Blau and Michaeli, 2019] is carefully managed.

Generator The auto-regressive LM generates audio tokens by conditioning on control signals that steer the generation towards the desired audio output. Each training example consists of paired annotations and audio, with the annotations converted into a sequence of embeddings that serves as the prefix for the LM. The handling of different control signal modalities is summarized as follows:

- Categorical signals: Closed-vocabulary tags (e.g., music genre) are converted into categorical embeddings using a lookup table, while free-form text descriptions are processed using a general-purpose text encoder from MuLan [Huang et al., 2022].
- Floating-point signals: Variables like melody note duration or song length are embedded using xVal encoding [Golkar et al., 2023] to represent continuous numerical inputs.
- Lyrics signals: Lyrics are transformed into phoneme sequences to capture pronunciation, improving the model's generalization to unseen words.
- Reference audio signals: The tokenizer extracts discrete token sequences from the reference audio, which are then mapped to continuous embeddings using a lookup table of the same size as the tokenizer's codebook, or further aggregated into track-level embeddings.

During training, the model minimizes cross-entropy loss on a next-token prediction task using teacher forcing. At inference, user inputs are converted into prefix embeddings based on the specified modalities, and the audio tokens are generated auto-regressively.

Renderer Once the auto-regressive LM generates the audio tokens, these tokens are processed by the Renderer to produce a rich, high-quality audio waveform. The Renderer is a cascaded system composed of two components: a Diffusion Transformer (DiT) and an acoustic vocoder, both of which are trained independently. The DiT employs a standard architecture with stacked blocks of attention layers and multi-layer perceptrons (MLPs). Its objective is to reverse the diffusion process, predicting clean vocoder latents from noise by estimating the noise level at each step. The acoustic vocoder is the decoder from a low frame-rate VAE vocoder and follows designs similar to [Kumar et al., 2024, Lee et al., 2022, Cong et al., 2021, Liu and Qian, 2021]. We found that structuring the vocoder latents as an information bottleneck within the cascaded system, combined with optimizing it with manageable model size and training time, results in superior audio quality and richer acoustic details compared to a single model that directly converts audio tokens into waveform.

3.2 Symbolic Token-based Pipeline

In contrast to the audio token-based pipeline, the symbolic token-based Generator, as shown in Figure 3, is designed to predict symbolic tokens for better interpretability, which is crucial for addressing musicians' workflows in Seed-Music.

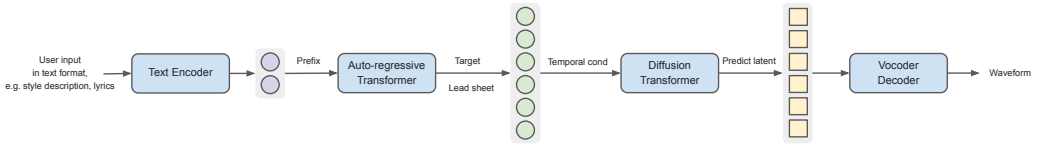

Figure 3. Overview of the pipeline using symbolic tokens as the intermediate representation. (1) Conditioned on the user prompt, the auto-regressive LM generates the symbolic tokens corresponding to a lead sheet. (2) The diffusion transformer model generates continuous vocoder latents given the symbolic tokens. (3) The vocoder then generates the high-quality 44KHz stereo audio waveform.

Prior efforts have proposed algorithms for melody generation [Ju et al., 2021, Zhang et al., 2022]. However, they lack explicit phoneme- and note-aligned information crucial for vocal music generation. Moreover, they remain limited to symbolic music generation without the capability for audio rendering. In a different line of research, there are task-specific prior works studying the approaches to steer music audio generation through musically interpretable conditions like harmony [Copet et al., 2024], dynamics, and rhythm [Wu et al., 2023]. Inspired by how jazz musicians use lead sheets to outline a composition's melody, harmony and structure, we introduce "lead sheet tokens" as a symbolic music representation. We highlight the key components, benefits, and limitations of lead sheet tokens compared to audio tokens below.

- To extract the symbolic features from audio for training the above system, we utilize in-house MIR models, including beat tracking [Hung et al., 2022], key and chord detection [Lu et al., 2021], structural section segmentation [Wang et al., 2022], five-instrument MIDI transcription (i.e., vocals, piano, guitar, bass, and drums) [Lu et al., 2023, Wang et al., 2024a], and singing lyrics transcription. The lead sheet tokens represent note-level details such as pitch, duration, position within a bar, vocal phonemes aligned to notes, and track-level attributes like section, instrument, key, and tempo.
- The one-to-one mapping between lead sheet tokens and human-readable lead sheets allows users to understand, edit, and interact with the musical scores directly. We experimented with different methods to generate lead sheet token sequences: REMI-style [Huang and Yang, 2020] and xVal [Golkar et al., 2023]. The REMI-style method interleaves instrument tracks into a quantized beat-based format, while xVal encodes onset and duration as continuous values. Although xVal-style encoding better follows our generative model's end product, the music performance, more closely, we found that the REMI-style one better suited for user interaction with musicians.
- Lead sheet tokens allow for the incorporation of human knowledge during both training and inference. For instance, music theory rules can be applied as constraints when predicting the next token in the sequence to enhance prediction accuracy.
- As the lead sheet tokens lack acoustic feature characterization, we scale up the token-to-latent diffusion model in the cascaded renderer to achieve the same end-to-end performance as the audio token-based system.

3.3 Vocoder Latent-based Pipeline

Previous works [Evans et al., 2024c,d, Levy et al., 2023, Rombach et al., 2022] have shown that an efficient approach for the task of "text-to-music" is to directly predict the vocoder latents using a latent diffusion model. Similarly, we train a variational autoencoder (VAE) operating at a low latent frame rate, alongside a diffusion transformer (DiT) that maps conditional inputs to normalized, continuous vocoder latents, as illustrated in Figure 4.

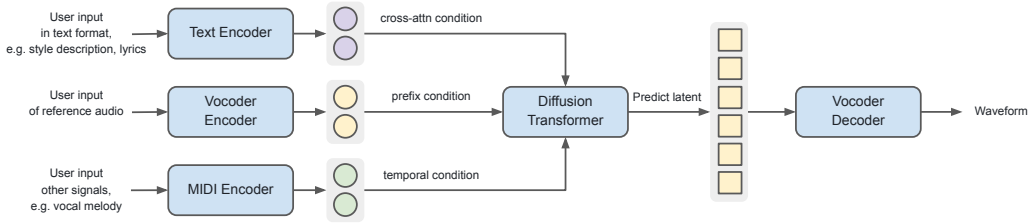

Figure 4. Seed-Music pipeline with vocoder latents as intermediate representation. (1) Various input types are fed into DiT via cross-attention, prefix, or time-aligned conditioning. (2) The diffusion transformer model predicts the continuous vocoder latents. (3) The acoustic vocoder then produces high-quality 44.1kHz stereo audio.

Compared to the audio token-based pipeline (see Section 3.1), the auto-regressive transformer module is omitted, although the architectures of the DiT and vocoder remain largely similar. To achieve comparable performance, the model size of each remaining module is scaled up. In the auto-regressive approach, all conditioning inputs are encoded into tokens in the prefix sequence, which can result in an excessively long prefix that degrades performance when handling larger and more diverse inputs. On the contrary, the vocoder latent-based design offers greater flexibility for incorporating a wider range of conditioning signals and supporting multi-channel inputs and outputs [Huang et al., 2023a]. We summarized how different types of prompts are used as follows:

- In-context conditioning in the vocoder latent space: This enables audio in-painting scenarios, such as audio continuation and editing.
- In-context conditioning in the input noise space [Peebles and Xie, 2023]: For variable-length inputs like lyrics and style descriptions, cross-attention layers are applied at each transformer block to incorporate these inputs.

- Time-aligned inputs that span multiple tracks: Temporal signals such as melody contours, intensity curves, and instrumental stems that are time-aligned conditioning inputs can be added at each step of the denoising process.
- Multi-channel outputs: Supported when multi-channel output examples are provided during training. For instance, the model can generate multiple musically distinct stems (e.g., vocals, bass, drums, and guitar), enabling downstream production scenarios like mashups and remixing. These stem-level training examples can be sourced from Music Source Separation (MSS) models.

3.4 Model Training and Inference

For all above mentioned pipelines, Seed-Music undergoes three training stages: pre-training, fine-tuning, and post-training similar to Seed-TTS and other text-based LMs. The pre-training stage aims to establish a better foundation for general music audio modeling. The fine-tuning stage consists of either data fine-tuning to enhance musicality, or instruction fine-tuning to improve controllability, interpretability, and interactivity for specific creation workflows.

Post-training of Seed-Music is conducted through Reinforcement Learning (RL), which has proven to be an effective learning paradigm in text and image processing [Schulman et al., 2017, Rafailov et al., 2024, Sutton et al., 1999, Esser et al., 2024b, Wallace et al., 2023]. Recent research has shown that Proximal Preference Optimization (PPO) can be extended to music and speech generation [Cideron et al., 2024, Zhang et al., 2024].

Inspired by these findings, we explore RL methods [Ahmadian et al., 2024, Prabhavalkar et al., 2018, Wang et al., 2024b, Sutton et al., 1999, Schulman et al., 2017] to improve the alignment of generated output with various input control signals and enhance musicality. Reward models we considered include: the edit distance between the original lyrics prompt and the lyrics transcription extracted from the generated audio, the genre prediction accuracy by comparing the input genre with the detected genre of the audio output, and the match between a song structure prompt and the detected structure in the generated audio. Additional reward models based on tempo, instrumentation, audio references, and singing voice references can be used to dictate what musical attributes are emphasized in the generation output. Moreover, incorporating human feedback [Ouyang et al., 2022] can produce reward models that capture subtle user preferences beyond the above objective metrics. We leave the thorough study of RL for future work.

During inference, the choice of sample decoding scheme plays a critical role in both output quality and stability for auto-regressive and diffusion models. We observed that carefully tuning classifier-free guidance [Ho and Salimans, 2022, Sanchez et al., 2023] is essential to ensure musicality and adherence to prompts. To reduce latency, we apply model distillation [Song et al., 2023] to minimize the number of iteration steps required by the DiT models. Additionally, we developed a streaming decoding scheme, allowing audio to be streamed while the auto-regressive model continues generating the token sequence.

4 Experiments

In this section, we showcase four applications powered by our model's capabilities: Lyrics2Song (Section 4.1), Lyrics2Leadsheet2Song (Section 4.2), MusicEDiT (Section 4.3), and Zero-shot Singing Voice Conversion (Section 4.4).

In **Lyrics2Song**, we introduce a vocal music generation system that produces performance-quality music with vocals based on user-provided lyrics and music style inputs. **Lyrics2Leadsheet2Song** builds on the Lyrics2Song system by incorporating symbolic music representation for enhanced interpretability. This process additionally generates a lead sheet, where users can access and adjust the melody and rhythm, allowing for finer control over the final audio output. **MusicEDiT** explores a diffusion-based in-painting system that enables users to edit the lyrics and melodies of an existing music audio piece. This serves as a post-production tool for modifying the vocals of a song. In **Zero-shot Singing Voice Conversion**, we offer a solution that allows users to modify the timbre of vocals in an existing audio based on their own voice with minimal reference data. This application facilitates vocal personalization with a low preparation cost. For each of these aforementioned applications,

we discuss the design choices related to intermediate representations, model architecture, and other configurations that optimize the system for its respective use case.

4.1 Lyrics2Song

Lyrics2Song generates vocal music performances conditioned on user-provided lyrics and music style descriptions. This task utilizes the audio token-based pipeline (see Section 3.1), leveraging tokenization and auto-regressive techniques to align multi-modal data (i.e., lyrics, style tags, and audio) and enable streaming decoding for fast, responsive interactions.

This system supports both cohesive short-form audio clip generation[2] and full-length track production[3]. The generated audio showcases expressive and dynamic vocal performances with engaging melodies as well as instrumentals that span a wide variety of genres and moods, delivering a mature level of musicality.

Vocal Music Generation with Audio Reference In addition to style descriptions, our system supports audio inputs to guide music generation. The listening examples[4] demonstrate how outputs are generated by referencing the musical styles of audio prompts. Since describing desired music with text or tags can be less intuitive for novice users, audio prompts provide a more effective way to communicate musical intent. Our system supports two modes of audio prompting. In "continuation mode," audio tokens extracted from the reference audio are concatenated in the prefix to continue auto-regressive generation, ensuring strong structural, melodic, and sonic similarities to the reference. In "remix mode," the input audio is converted into an embedding vector within a pre-trained joint text-audio embedding space [Huang et al., 2022]. This embedding, which summarizes the global characteristics of the reference audio, is then incorporated into the prefix to guide the generation, allowing the output audio to adopt different styles.

In both modes, we demonstrate the versatility of audio prompting with pairing different lyrics inputs for generation and the lyrics from the reference song. When the lyrics used for generation closely follow the original lyrics in the reference song, we observe seamless continuation of the reference, with echoing melody line and song structure. When the new lyrics differ significantly from the reference song in style—such as syllable count, structure, or language—the similarity between the generated result and the reference audio weakens. Despite this, there remains a natural resemblance in musical motifs, rhythm patterns, instrumentation, voice timbre, and overall vibe.

Instrumental Music Generation The same pipeline supports pure instrumental music generation, where the creator omits the lyrics and simply provides descriptors for the desired music. We provide audio samples[5] in a wide range of styles with creative musical transitions to fit precise timing prompts for section transitions.

Evaluation Metrics During model development, we considered the following quantitative metrics, which are also repurposed as reward models in reinforcement learning of the audio token based LM.

- **WER**: We employ the Whisper-large-v3 model for English vocal music and the Paraformer-zh for Chinese vocal music to transcribe the generation example and compute the word (Pingyin) error rate and against the lyrics prompt. Although WER is a typical metric for TTS, it is not a perfect measure for vocal music quality. Sung words usually introduce elongated vowels and consonants as well as larger ranges in pitch and non speech-like cadences. These are attributes that an automatic WER could penalize but are critical for dynamic and expressive vocal music.
- **MIR classification**: To measure the adherence between generated audio and style prompt, we use in-house Music Information Retrieval (MIR) models to run inference on the generated audio, and check the correspondence between the predicted genre, mood, vocal timbre, song structure with timing information, against the music elements mentioned in the style prompt.

For qualitative metrics, we use the Comparative Mean Opinion Score (CMOS) based on a team of musically-trained raters. We define the following critical dimensions for assessment.

[2] https://team.doubao.com/seed-music/shortform-audio-generation
[3] https://team.doubao.com/seed-music/longform-audio-generation
[4] https://team.doubao.com/seed-music/audio-prompting
[5] https://team.doubao.com/seed-music/instrumental-music-generation

- **Musicality** measures musically semantic attributes and was decomposed into interesting vocal melodies, appropriate use of harmony, presence of idiomatic musical forms like theme and variation, plausible structure, suitable chord progressions, characteristic rhythmic patterns and full-bodied instrumentation arrangement.
- **Audio Quality** measures acoustic attributes including the clarity of vocals, realism of instruments, detail across the frequency spectrum and transients of drums and onsets. Raters also consider unwanted audio artifacts like distortion, muffled audio or missing frequency bands.
- **Prompt Adherence** measures how closely the generated audio matches the lyrics input and other style prompts.

Unlike in speech domains, where Word Error Rate (WER) and Automatic Speaker Verification (ASV) are widely used to compare TTS and ASR models, no such straightforward quantitative metric or benchmark datasets exist in music. Moreover, musicality, the most important aspect, is nearly impossible to be quantified by an objective metric. We hope our documented methodology can guide subsequent research in vocal music generation. We note the challenge in presenting quantitative and qualitative statistics that can be compared against for music generation, and thus we encourage readers to directly listen to the many audio demos on the website to judge the quality of our system.

We have also conducted Lyrics2Song experiments with a diffusion-based pipeline as outlined in Section 3.3, and achieved on-par end-to-end performance. Nevertheless, we find the LM-based pipeline to be naturally suited for interactive use cases. Its causal nature enables streaming solutions like those described in Section 3.1 to provide a near real-time experience and allows future integration with multi-modal models [Liu et al., 2023].

4.2 Lyrics2Leadsheet2Song

In this set of experiments, we develop solutions around the system pipeline depicted in Section 3.2 to expose lead sheet tokens for musicians to interact with as an intermediate step in the end-to-end music audio generation process.

Lyrics2Leadsheet We developed a rule-based symbolic music codec based on [Chen et al., 2024c] to encode the symbolic features of a music audio piece into a sequence of lead sheet tokens. In Figure 5, we visualize how the information of different tracks, including lyrics, vocal melody, piano, guitar, drum, chord, are encoded in the token sequence. We interleave different tracks on a bar-by-bar basis; and for individual element (e.g. phoneme, note and chord) in each track, we encode the onset, duration, and pitch information if available.

Figure 5: Illustration of the REMI-style symbolic music encoding.

We show with the demo examples[6] how the LM Generator, trained with the transcribed lead sheets with the MIR models discussed in Section 3.2, learns to predict phoneme-aligned notes and genre-appropriate melodies and rhythms based on the lyrics and style prompts.

Leadsheet2Song Leadsheet2Song generalizes the task of transforming symbolic representations into full mixes of music audio. By inspecting the listening examples[7], one can hear how the generated audio follows vocal melodies, phonemes, instrumental melodies, rhythms and chords specified in the lead sheet tokens, and renders the full mixture of natural and expressive music performance.

[6] https://team.doubao.com/seed-music/lyrics-to-leadsheet
[7] https://team.doubao.com/seed-music/leadsheet-to-song

Leadsheet2Vocals (Singing Voice Synthesis) Instead of outputting a full mix, the lead sheet token based generative system can be configured to generate individual stems. Singing Voice Synthesis (SVS) becomes a straightforward subset of lead sheet token-based pipelines as shown in the demo[8].

4.3 Music Editing

In this section, we consider the task of making edits to music audio as a post-production process. The non-causal nature of diffusion-based approach proposed in Section 3.3 makes it naturally suited for editing tasks. For example, in text-conditioned in-painting, a diffusion module's visibility of context before and after the masked audio segments guarantees smoother transitions [Wang et al., 2023c]. We formulate the problem as a lead sheet conditioned in-painting task to train the Diffusion Tranformer model. Upon inference, the modified lead sheet is used as conditioning input, and the segment of audio corresponding to the modified part of the lead sheet is masked and re-generated.

The listening examples[9] show the system's ability to precisely alter the sung lyrics while keeping the melody and backing track unchanged. Conversely, creators can also precisely alter the melody while keeping the melody and backing track unchanged[10]. We demonstrate this for English and Mandarin Chinese. We are particularly excited by this new form of "generative audio editing", which retains the natural performance and musical semantics of the original track. These are post-production edits that historically would have required original project stems and MIDI tracks.

4.4 Singing Voice Conversion

In the context of Seed-Music, singing voice conversion (SVC) is the final piece of the suite that bridges a creator's own voice with the generative systems. Although our VC system is similar to Seed-TTS, vocal music VC presents new challenges distinct from speech VC [Arik et al., 2018, Anastassiou et al., 2024].

- **Vocal mixture**: Vocal music data natively consists of vocal track and background music, whereas speech data has no background music. Although modern MSS algorithms can be applied and the resulting vocal stem treated with conventional voice conversion techniques, this approach is not immune from the artifacts introduced in the MSS process. Our goal was to develop a scalable system that could directly operate on the mix of vocal track and background music.
- **Vocal range**: The range of vocal singing is much wider than vocal speech. In the context of zero-shot singing voice conversion, the system must robustly extrapolate the synthesized singing voice to fundamental frequencies beyond the reference vocal.
- **Vocal technique**: Vocal singing techniques are much more complex than speech. The same vocalist can sound very different when singing in operatic mezzo-soprano style, musical theater belting style or jazz scatting style. Singing VC has to deliver an expressive rendition of these techniques on top of regular speech attributes like clear pronunciation and prosody.
- **Vocal vs speech reference**: In speech VC, it is a straightforward task to obtain reference speech from an end-user. In singing VC however, it is non-trivial for the end-user to provide a reference singing vocal performance. Since our goal was to creatively empower normal individuals, our system was deliberately designed to be no-singing-required. It can produce vocal music outputs given only a short *speech* clip as reference.
- **Amateur vs professional singing**: Speech benefits from a relatively easy procedure for finding paired data between an input speaker and target speaker e.g. recording different voices reading the same passage. However, no such large scale data exists for modeling an amateur singer and professional singer. A vocal music VC system must translate non-professional singing into an improved performance, e.g. if the end-user provides out-of-tune singing, the VC result should retain the tone of voice but actually be in tune.

Naturally, when the input reference vocal and target conversion vocal are similar, i.e. both are male voices singing in English, the results are the best. Other combinations are more challenging, such as when the input reference vocal is a female *speaking* (not singing) in Mandarin and the target

[8] https://team.doubao.com/seed-music/leadsheet-to-vocals
[9] https://team.doubao.com/seed-music/editing-lyrics
[10] https://team.doubao.com/seed-music/editing-melody

conversion vocal is a female singing in Mandarin or even harder still, singing in English. The listening examples[11] demonstrate how our VC system is able to handle these combinations.

5 Conclusion

In this report, we introduced Seed-Music, a family of generative models for music creation tasks. We demonstrated the generation of high quality vocal music conditioned on natural language style description, or audio reference, or music score. We proposed an interpretable lead sheet tokenizer, which empowered precise audio editing workflows. Our approach revolves around three intermediate representations, each chosen to best support musical workflows from ideation to generation and editing. The flexible nature of the framework opens the door to many exciting future works, including more granular stem generation and multi-modal conditioning signals.

Seed-Music lowers the barriers to artistic creation and musical expression. We strongly believe the demos shown in this report can empower creators across the spectrum of novices and professionals. For example, our coupling of text-to-music pipelines with zero-shot singing voice conversion was intended to integrate novices more deeply into the creative process. Rather than manipulating music from a distance, novices bring their own unique voices and identities to participate in creative ideation.

Music is also integral to many complementary mediums including shortform video, longform cinema, games and AR/VR experiences. Real-time conditioning and rendering of generative music opens up entirely new interactions beyond traditional playback of audio files. We imagine entirely new artistic mediums where generative music responds to conditioning signals beyond text but in-game storylines and visual art-style.

For professionals, our lead sheet tokens were engineered from the ground-up to respect and integrate into workflows of musicians, composers, vocalists and artists. We believe lead sheet tokens can evolve to become the symbolic standard for music language models, similar to today's industry-standard MIDI for traditional music making. Musicians and producers would be able to leverage the power of generative models through familiar means of control over melodic and rhythmic elements. Moreover, the ability to edit and manipulate recorded music in ways that preserve music semantics is a meaningful time-saving and cost-saving capability across the industry. We are also excited by future stem-based generation and editing workflows beyond the vocal track. Professionals can audition musical ideas with greater speed and experience higher chances of landing on "happy accidents" crucial to the creative process.

[11] https://team.doubao.com/seed-music/singing-voice-conversion

6 Ethics and Safety

We firmly believe that AI technologies should support, not disrupt, the livelihoods of musicians and artists. AI should serve as a tool for artistic expression, as true art always stems from human intention. Our goal is to present this technology as an opportunity to advance the music industry by lowering barriers to entry, offering smarter, faster editing tools, generating new and exciting sounds, and opening up new possibilities for artistic exploration.

Ethics We recognize that AI tools are inherently prone to bias, and our goal is to provide a tool that stays neutral and benefits everyone. To achieve this, we aim to offer a wide range of control elements that help minimize preexisting biases. By returning artistic choices to users, we believe we can promote equality, preserve creativity, and enhance the value of their work. With these priorities in mind, we hope our breakthroughs in lead sheet tokens highlight our commitment to empowering musicians and fostering human creativity through AI.

Safety In the case of vocal music, we recognize how the singing voice evokes one of the strongest expressions of individual identity. To safeguard against the misuse of this technology in impersonating others, we adopt a process similar to the safety measures laid out in Seed-TTS. This involves a multi-step verification method for spoken content and voice to ensure the enrollment of audio tokens contains only the voice of authorized users. We also implement a multi-level water-marking scheme and duplication checks across the generative process.

Modern systems for music generation may fundamentally reshape culture and the relationship between artistic creation and consumption. We are confident that, with strong consensus between stakeholders, these technologies will and revolutionize music creation workflow and benefit music novices, professionals, and listeners alike.

7 Acknowledgement

Authors (Alphabetical Order)

Ye Bai	Qingqing Huang	Xingxing Li	Ju-Chiang Wang	Yilin Zhang
Haonan Chen	Zhongyi Huang	Shouda Liu	Yuping Wang	Hang Zhao
Jitong Chen	Dongya Jia	Wei-Tsung Lu	Yuxuan Wang	Ziyi Zhao
Zhuo Chen	Feihu La	Yiqing Lu	Ling Xu	Dejian Zhong
Yi Deng	Duc Le	Andrew Shaw	Yifeng Yang	Shicen Zhou
Xiaohong Dong	Bochen Li	Janne Spijkervet	Chao Yao	Pei Zou
Lamtharn Hantrakul	Chumin Li	Yakun Sun	Shuo Zhang	
Weituo Hao	Hui Li	Bo Wang	Yang Zhang	

We extend our gratitude to the larger Seed team whose dedication and expertise were vital to the success of this project. Special thanks to our engineering team for their technical prowess; our data teams, whose diligent efforts in data collection, annotation, and processing were indispensable; our project operation team for providing logistical guidance; our evaluation team for their rigorous testing and insightful feedback; and also the Seed-TTS and Seed-ASR teams for valuable knowledge sharing. Their contributions have been instrumental to Seed-Music (no pun intended).

References

Samuel Mehr, Manvir Singh, Dean Knox, Daniel Ketter, Daniel Pickens-Jones, S Atwood, Christopher Lucas, Nori Jacoby, Alena Egner, Erin Hopkins, Rhea Howard, Joshua Hartshorne, Mariela Jennings, Jan Simson, Constance Bainbridge, Steven Pinker, Timothy O'Donnell, Max Krasnow, and Luke Glowacki. Universality and diversity in human song. *Science (New York, N.Y.)*, 366, 11 2019. doi: 10.1126/science.aax0868.

Bobby Owsinski. *The music producer's handbook*. Hal Leonard Corporation, 2010.

Mike Senior. *Mixing Secrets*. Routledge, 2012.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. A survey on multimodal large language models. *arXiv preprint arXiv:2306.13549*, 2023.

Jun Zhan, Junqi Dai, Jiasheng Ye, Yunhua Zhou, Dong Zhang, Zhigeng Liu, Xin Zhang, Ruibin Yuan, Ge Zhang, Linyang Li, Hang Yan, Jie Fu, Tao Gui, Tianxiang Sun, Yugang Jiang, and Xipeng Qiu. Anygpt: Unified multimodal llm with discrete sequence modeling, 2024. URL https://arxiv.org/abs/2402.12226.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners, 2019.

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, 2024. URL https://arxiv.org/abs/2406.02430.

William Peebles and Saining Xie. Scalable diffusion models with transformers, 2023. URL https://arxiv.org/abs/2212.09748.

Andrea Agostinelli, Timo I. Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, Matt Sharifi, Neil Zeghidour, and Christian Frank. Musiclm: Generating music from text, 2023. URL https://arxiv.org/abs/2301.11325.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation, 2024. URL https://arxiv.org/abs/2306.05284.

Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A generative model for music, 2020. URL https://arxiv.org/abs/2005.00341.

David Cope. Experiments in musical intelligence (emi): Non-linear linguistic-based composition. *Interface*, 18(1-2):117–139, 1989. doi: 10.1080/09298218908570541. URL https://doi.org/10.1080/09298218908570541.

Bob L. Sturm, João Felipe Santos, Oded Ben-Tal, and Iryna Korshunova. Music transcription modelling and composition using deep learning, 2016. URL https://arxiv.org/abs/1604.08723.

Ian Simon and Sageev Oore. Performance rnn: Generating music with expressive timing and dynamics. https://magenta.tensorflow.org/performance-rnn, 2017.

Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Noam Shazeer, Ian Simon, Curtis Hawthorne, Andrew M. Dai, Matthew D. Hoffman, Monica Dinculescu, and Douglas Eck. Music transformer, 2018. URL https://arxiv.org/abs/1809.04281.

OpenAI. Musenet. URL https://openai.com/index/musenet/.

Cheng-Zhi Anna Huang, Tim Cooijmans, Adam Roberts, Aaron Courville, and Douglas Eck. Counterpoint by convolution, 2019. URL https://arxiv.org/abs/1903.07227.

Hao-Wen Dong, Ke Chen, Julian McAuley, and Taylor Berg-Kirkpatrick. Muspy: A toolkit for symbolic music generation. *arXiv preprint arXiv:2008.01951*, 2020.

Pedro Sarmento, Adarsh Kumar, CJ Carr, Zack Zukowski, Mathieu Barthet, and Yi-Hsuan Yang. Dadagp: A dataset of tokenized guitarpro songs for sequence models. *arXiv preprint arXiv:2107.14653*, 2021.

Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Douglas Eck, Karen Simonyan, and Mohammad Norouzi. Neural audio synthesis of musical notes with wavenet autoencoders, 2017. URL https://arxiv.org/abs/1704.01279.

Jesse Engel, Kumar Krishna Agrawal, Shuo Chen, Ishaan Gulrajani, Chris Donahue, and Adam Roberts. Gansynth: Adversarial neural audio synthesis, 2019. URL https://arxiv.org/abs/1902.08710.

Jesse Engel, Lamtharn Hantrakul, Chenjie Gu, and Adam Roberts. Ddsp: Differentiable digital signal processing, 2020. URL https://arxiv.org/abs/2001.04643.

Yusong Wu, Ethan Manilow, Yi Deng, Rigel Swavely, Kyle Kastner, Tim Cooijmans, Aaron Courville, Cheng-Zhi Anna Huang, and Jesse Engel. Midi-ddsp: Detailed control of musical performance via hierarchical modeling, 2022. URL https://arxiv.org/abs/2112.09312.

Nal Kalchbrenner, Erich Elsen, Karen Simonyan, Seb Noury, Norman Casagrande, Edward Lockhart, Florian Stimberg, Aaron van den Oord, Sander Dieleman, and Koray Kavukcuoglu. Efficient neural audio synthesis, 2018. URL https://arxiv.org/abs/1802.08435.

Lamtharn (Hanoi) Hantrakul, Jesse Engel, Adam Roberts, and Chenjie Gu. Fast and flexible neural audio synthesis, 2019. URL https://archives.ismir.net/ismir2019/paper/000063.pdf.

Xavier Serra and Julius Smith. Spectral modeling synthesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition. *Computer Music Journal*, 14(4):12–24, 1990. ISSN 01489267, 15315169. URL http://www.jstor.org/stable/3680788.

Curtis Hawthorne, Andriy Stasyuk, Adam Roberts, Ian Simon, Cheng-Zhi Anna Huang, Sander Dieleman, Erich Elsen, Jesse Engel, and Douglas Eck. Enabling factorized piano music modeling and generation with the MAESTRO dataset. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=r1lYRjC9F7.

Perry R Cook. Singing voice synthesis: History, current work, and future directions. *Computer Music Journal*, 20(3):38–46, 1996.

Masanari Nishimura, Kei Hashimoto, Keiichiro Oura, Yoshihiko Nankaku, and Keiichi Tokuda. Singing voice synthesis based on deep neural networks. In *Interspeech*, pages 2478–2482, 2016.

Yuan-Hao Yi, Yang Ai, Zhen-Hua Ling, and Li-Rong Dai. Singing voice synthesis using deep autoregressive neural networks for acoustic modeling. *arXiv preprint arXiv:1906.08977*, 2019.

Peiling Lu, Jie Wu, Jian Luan, Xu Tan, and Li Zhou. Xiaoicesing: A high-quality and integrated singing voice synthesis system. *arXiv preprint arXiv:2006.06261*, 2020.

Xiaobin Zhuang, Tao Jiang, Szu-Yu Chou, Bin Wu, Peng Hu, and Simon Lui. LiteSing: Towards fast, lightweight and expressive singing voice synthesis. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7078–7082. IEEE, 2021.

Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, and Neil Zeghidour. Audiolm: a language modeling approach to audio generation, 2023. URL https://arxiv.org/abs/2209.03143.

Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507, 2021.

Ye Bai, Jingping Chen, Jitong Chen, Wei Chen, Zhuo Chen, Chuang Ding, Linhao Dong, Qianqian Dong, Yujiao Du, Kepan Gao, Lu Gao, Yi Guo, Minglun Han, Ting Han, Wenchao Hu, Xinying Hu, Yuxiang Hu, Deyu Hua, Lu Huang, Mingkun Huang, Youjia Huang, Jishuo Jin, Fanliu Kong, Zongwei Lan, Tianyu Li, Xiaoyang Li, Zeyang Li, Zehua Lin, Rui Liu, Shouda Liu, Lu Lu, Yizhou Lu, Jingting Ma, Shengtao Ma, Yulin Pei, Chen Shen, Tian Tan, Xiaogang Tian, Ming Tu, Bo Wang, Hao Wang, Yuping Wang, Yuxuan Wang, Hanzhang Xia, Rui Xia, Shuangyi Xie, Hongmin Xu, Meng Yang, Bihong Zhang, Jun Zhang, Wanyi Zhang, Yang Zhang, Yawei Zhang, Yijie Zheng, and Ming Zou. Seed-asr: Understanding diverse speech and contexts with llm-based speech recognition, 2024. URL https://arxiv.org/abs/2407.04675.

Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, Lei He, Sheng Zhao, and Furu Wei. Neural codec language models are zero-shot text to speech synthesizers, 2023a. URL https://arxiv.org/abs/2301.02111.

Ziqiang Zhang, Long Zhou, Chengyi Wang, Sanyuan Chen, Yu Wu, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, Lei He, Sheng Zhao, and Furu Wei. Speak foreign languages with your own voice: Cross-lingual neural codec language modeling, 2023. URL https://arxiv.org/abs/2303.03926.

Sanyuan Chen, Shujie Liu, Long Zhou, Yanqing Liu, Xu Tan, Jinyu Li, Sheng Zhao, Yao Qian, and Furu Wei. Vall-e 2: Neural codec language models are human parity zero-shot text to speech synthesizers, 2024a. URL https://arxiv.org/abs/2406.05370.

James Betker. Better speech synthesis through scaling. *arXiv preprint arXiv:2305.07243*, 2023.

Mateusz Łajszczak, Guillermo Cámbara, Yang Li, Fatih Beyhan, Arent van Korlaar, Fan Yang, Arnaud Joly, Álvaro Martín-Cortinas, Ammar Abbas, Adam Michalski, et al. BASE TTS: Lessons from building a billion-parameter text-to-speech model on 100k hours of data. *arXiv preprint arXiv:2402.08093*, 2024.

Qingqing Huang, Aren Jansen, Joonseok Lee, Ravi Ganti, Judith Yue Li, and Daniel PW Ellis. Mulan: A joint embedding of music audio and natural language. *arXiv preprint arXiv:2208.12415*, 2022.

Yu-An Chung, Yu Zhang, Wei Han, Chung-Cheng Chiu, James Qin, Ruoming Pang, and Yonghui Wu. W2v-bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training, 2021. URL https://arxiv.org/abs/2108.06209.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Ruiqi Li, Rongjie Huang, Yongqi Wang, Zhiqing Hong, and Zhou Zhao. Self-supervised singing voice pre-training towards speech-to-singing conversion. *arXiv preprint arXiv:2406.02429*, 2024a.

Ruiqi Li, Zhiqing Hong, Yongqi Wang, Lichao Zhang, Rongjie Huang, Siqi Zheng, and Zhou Zhao. Accompanied singing voice synthesis with fully text-controlled melody, 2024b. URL https://arxiv.org/abs/2407.02049.

Rongjie Huang, Chunlei Zhang, Yongqi Wang, Dongchao Yang, Jinchuan Tian, Luping Liu, Zhenhui Ye, Ziyue Jiang, Xuankai Chang, Jiatong Shi, CHAO WENG, Zhou Zhao, and Dong Yu. MVoice: Multilingual unified voice generation with discrete representation at scale, 2024. URL https://openreview.net/forum?id=eGdhD93hZr.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. URL https://arxiv.org/abs/2006.11239.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. URL https://arxiv.org/abs/2010.02502.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021. URL https://arxiv.org/abs/2011.13456.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. URL https://arxiv.org/abs/2112.10752.

Qingqing Huang, Daniel S. Park, Tao Wang, Timo I. Denk, Andy Ly, Nanxin Chen, Zhengdong Zhang, Zhishuai Zhang, Jiahui Yu, Christian Frank, Jesse Engel, Quoc V. Le, William Chan, Zhifeng Chen, and Wei Han. Noise2music: Text-conditioned music generation with diffusion models, 2023a. URL https://arxiv.org/abs/2302.03917.

Zach Evans, CJ Carr, Josiah Taylor, Scott H. Hawley, and Jordi Pons. Fast timing-conditioned latent audio diffusion, 2024a. URL https://arxiv.org/abs/2402.04825.

Zach Evans, Julian D. Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. Stable audio open, 2024b. URL https://arxiv.org/abs/2407.14358.

Jan Melechovsky, Zixun Guo, Deepanway Ghosal, Navonil Majumder, Dorien Herremans, and Soujanya Poria. Mustango: Toward controllable text-to-music generation, 2024. URL https://arxiv.org/abs/2311.08355.

Ke Chen, Yusong Wu, Haohe Liu, Marianna Nezhurina, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Musicldm: Enhancing novelty in text-to-music generation using beat-synchronous mixup strategies, 2023. URL https://arxiv.org/abs/2308.01546.

Haohe Liu, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Qiao Tian, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D. Plumbley. Audioldm 2: Learning holistic audio generation with self-supervised pretraining, 2024. URL https://arxiv.org/abs/2308.05734.

Shih-Lun Wu, Chris Donahue, Shinji Watanabe, and Nicholas J. Bryan. Music controlnet: Multiple time-varying controls for music generation, 2023. URL https://arxiv.org/abs/2311.07069.

Or Tal, Alon Ziv, Itai Gat, Felix Kreuk, and Yossi Adi. Joint audio and symbolic conditioning for temporally controlled text-to-music generation, 2024. URL https://arxiv.org/abs/2406.10970.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021.

Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance, 2022. URL https://arxiv.org/abs/2210.11427.

Xuan Su, Jiaming Song, Chenlin Meng, and Stefano Ermon. Dual diffusion implicit bridges for image-to-image translation, 2023. URL https://arxiv.org/abs/2203.08382.

Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models, 2023. URL https://arxiv.org/abs/2210.09276.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis, 2024a. URL https://arxiv.org/abs/2403.03206.

Jinglin Liu, Chengxi Li, Yi Ren, Feiyang Chen, and Zhou Zhao. Diffsinger: Singing voice synthesis via shallow diffusion mechanism, 2022. URL https://arxiv.org/abs/2105.02446.

Hui Li, Hongyu Wang, Zhijin Chen, Bohan Sun, and Bo Li. Real-time and accurate: Zero-shot high-fidelity singing voice conversion with multi-condition flow synthesis. *arXiv preprint arXiv:2405.15093*, 2024c.

Shihao Chen, Yu Gu, Jie Zhang, Na Li, Rilin Chen, Liping Chen, and Lirong Dai. Ldm-svc: Latent diffusion model based zero-shot any-to-any singing voice conversion with singer guidance. *arXiv preprint arXiv:2406.05325*, 2024b.

Ryuichi Yamamoto, Reo Yoneyama, Lester Phillip Violeta, Wen-Chin Huang, and Tomoki Toda. A comparative study of voice conversion models with large-scale speech and singing data: The t13 systems for the singing voice conversion challenge 2023. In *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1–6. IEEE, 2023.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. Clap: Learning audio concepts from natural language supervision, 2022. URL https://arxiv.org/abs/2206.04769.

Kainan Peng, Wei Ping, Zhao Song, and Kexin Zhao. Non-autoregressive neural text-to-speech, 2020. URL https://arxiv.org/abs/1905.08459.

Marco Pasini, Stefan Lattner, and George Fazekas. Music2latent: Consistency autoencoders for latent audio compression, 2024. URL https://arxiv.org/abs/2408.06500.

Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High fidelity neural audio compression, 2022. URL https://arxiv.org/abs/2210.13438.

Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-fidelity audio compression with improved rvqgan, 2023. URL https://arxiv.org/abs/2306.06546.

Dongchao Yang, Songxiang Liu, Rongjie Huang, Jinchuan Tian, Chao Weng, and Yuexian Zou. Hifi-codec: Group-residual vector quantization for high fidelity audio codec, 2023. URL https://arxiv.org/abs/2305.02765.

Shengpeng Ji, Ziyue Jiang, Xize Cheng, Yifu Chen, Minghui Fang, Jialong Zuo, Qian Yang, Ruiqi Li, Ziang Zhang, Xiaoda Yang, Rongjie Huang, Yidi Jiang, Qian Chen, Siqi Zheng, Wen Wang, and Zhou Zhao. Wavtokenizer: an efficient acoustic discrete codec tokenizer for audio language modeling, 2024. URL https://arxiv.org/abs/2408.16532.

Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. Masked autoencoders that listen, 2023b. URL https://arxiv.org/abs/2207.06405.

Matthew Le, Apoorv Vyas, Bowen Shi, Brian Karrer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, et al. Voicebox: Text-guided multilingual universal speech generation at scale. *Advances in neural information processing systems*, 36, 2024.

Apoorv Vyas, Bowen Shi, Matthew Le, Andros Tjandra, Yi-Chiao Wu, Baishan Guo, Jiemin Zhang, Xinyue Zhang, Robert Adkins, William Ngan, Jeff Wang, Ivan Cruz, Bapi Akula, Akinniyi Akinyemi, Brian Ellis, Rashel Moritz, Yael Yungster, Alice Rakotoarison, Liang Tan, Chris Summers, Carleigh Wood, Joshua Lane, Mary Williamson, and Wei-Ning Hsu. Audiobox: Unified audio generation with natural language prompts, 2023. URL https://arxiv.org/abs/2312.15821.

Chengyi Wang, Sanyuan Chen, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, Lei He, Sheng Zhao, and Furu Wei. Neural codec language models are zero-shot text to speech synthesizers, 2023b.

Yochai Blau and Tomer Michaeli. Rethinking lossy compression: The rate-distortion-perception tradeoff. In *International Conference on Machine Learning*, pages 675–685. PMLR, 2019.

Siavash Golkar, Mariel Pettee, Michael Eickenberg, Alberto Bietti, Miles Cranmer, Geraud Krawezik, Francois Lanusse, Michael McCabe, Ruben Ohana, Liam Parker, et al. xval: A continuous number encoding for large language models. *arXiv preprint arXiv:2310.02989*, 2023.

Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-fidelity audio compression with improved RVQGAN. *Advances in Neural Information Processing Systems*, 36, 2024.

Sang-gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon. BigVGAN: A universal neural vocoder with large-scale training. *arXiv preprint arXiv:2206.04658*, 2022.

Jian Cong, Shan Yang, Lei Xie, and Dan Su. Glow-WaveGAN: Learning speech representations from gan-based variational auto-encoder for high fidelity flow-based speech synthesis. *arXiv preprint arXiv:2106.10831*, 2021.

Zhengxi Liu and Yanmin Qian. Basis-MelGAN: Efficient neural vocoder based on audio decomposition. *arXiv preprint arXiv:2106.13419*, 2021.

Zeqian Ju, Peiling Lu, Xu Tan, Rui Wang, Chen Zhang, Songruoyao Wu, Kejun Zhang, Xiangyang Li, Tao Qin, and Tie-Yan Liu. Telemelody: Lyric-to-melody generation with a template-based two-stage method. *arXiv preprint arXiv:2109.09617*, 2021.

Daiyu Zhang, Ju-Chiang Wang, Katerina Kosta, Jordan BL Smith, and Shicen Zhou. Modeling the rhythm from lyrics for melody generation of pop song. In *Proc. ISMIR*, pages 141–148, 2022.

Yun-Ning Hung, Ju-Chiang Wang, Xuchen Song, Wei-Tsung Lu, and Minz Won. Modeling beats and downbeats with a time-frequency transformer. In *ICASSP*, pages 401–405, 2022.

Wei-Tsung Lu, Ju-Chiang Wang, Minz Won, Keunwoo Choi, and Xuchen Song. SpecTNT: A time-frequency transformer for music audio. In *ISMIR*, 2021.

Ju-Chiang Wang, Yun-Ning Hung, and Jordan BL Smith. To catch a chorus, verse, intro, or anything else: Analyzing a song with structural functions. In *ICASSP*, pages 416–420, 2022.

Wei-Tsung Lu, Ju-Chiang Wang, and Yun-Ning Hung. Multitrack music transcription with a time-frequency perceiver. In *ICASSP*, 2023.

Ju-Chiang Wang, Wei-Tsung Lu, and Jitong Chen. Mel-RoFormer for vocal separation and vocal melody transcription. In *ISMIR*, 2024a.

Yu-Siang Huang and Yi-Hsuan Yang. Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions. In *Proceedings of the 28th ACM international conference on multimedia*, pages 1180–1188, 2020.

Zach Evans, Julian D Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. Long-form music generation with latent diffusion. *arXiv preprint arXiv:2404.10301*, 2024c.

Zach Evans, CJ Carr, Josiah Taylor, Scott H Hawley, and Jordi Pons. Fast timing-conditioned latent audio diffusion. *arXiv preprint arXiv:2402.04825*, 2024d.

Mark Levy, Bruno Di Giorgi, Floris Weers, Angelos Katharopoulos, and Tom Nickson. Controllable music production with diffusion models and guidance gradients. *arXiv preprint arXiv:2311.00613*, 2023.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12, 1999.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. *arXiv preprint arXiv:2403.03206*, 2024b.

Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. *arXiv preprint arXiv:2311.12908*, 2023.

Geoffrey Cideron, Sertan Girgin, Mauro Verzetti, Damien Vincent, Matej Kastelic, Zalán Borsos, Brian McWilliams, Victor Ungureanu, Olivier Bachem, Olivier Pietquin, et al. MusicRL: Aligning music generation to human preferences. *arXiv preprint arXiv:2402.04229*, 2024.

Dong Zhang, Zhaowei Li, Shimin Li, Xin Zhang, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. SpeechAlign: Aligning speech generation to human preferences. *arXiv preprint arXiv:2404.05600*, 2024.

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Ahmet Üstün, and Sara Hooker. Back to basics: Revisiting REINFORCE style optimization for learning from human feedback in LLMs. *arXiv preprint arXiv:2402.14740*, 2024.

Rohit Prabhavalkar, Tara N Sainath, Yonghui Wu, Patrick Nguyen, Zhifeng Chen, Chung-Cheng Chiu, and Anjuli Kannan. Minimum word error rate training for attention-based sequence-to-sequence models. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4839–4843. IEEE, 2018.

Zihao Wang, Chirag Nagpal, Jonathan Berant, Jacob Eisenstein, Alex D'Amour, Sanmi Koyejo, and Victor Veitch. Transforming and combining rewards for aligning large language models. *arXiv preprint arXiv:2402.00742*, 2024b.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

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

Guillaume Sanchez, Honglu Fan, Alexander Spangher, Elad Levi, Pawan Sasanka Ammanamanchi, and Stella Biderman. Stay on topic with classifier-free guidance. *arXiv preprint arXiv:2306.17806*, 2023.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.

Shansong Liu, Atin Sakkeer Hussain, Chenshuo Sun, and Ying Shan. Music understanding llama: Advancing text-to-music generation with question answering and captioning, 2023. URL https://arxiv.org/abs/2308.11276.

Haonan Chen, Jordan B. L. Smith, Bochen Li, Ju-Chiang Wang, Janne Spijkervet, Pei Zou, Xingjian Du, and Qiuqiang Kong. SymPAC: Scalable symbolic music generation with prompts and constraints. In *ISMIR*, 2024c.

Yuancheng Wang, Zeqian Ju, Xu Tan, Lei He, Zhizheng Wu, Jiang Bian, and Sheng Zhao. Audit: Audio editing by following instructions with latent diffusion models, 2023c. URL https://arxiv.org/abs/2304.00830.

Sercan Arik, Jitong Chen, Kainan Peng, Wei Ping, and Yanqi Zhou. Neural voice cloning with a few samples. *Advances in neural information processing systems*, 31, 2018.