Analyzable Chain-of-Musical-Thought Prompting for High-Fidelity Music Generation

Max W. Y. Lam[*†], **Yijin Xing**[*], **Weiya You, Jingcheng Wu, Zongyu Yin, Fuqiang Jiang, Hangyu Liu, Feng Liu, Xingda Li, Wei-Tsung Lu, Hanyu Chen, Tong Feng, Tianwei Zhao, Chien-Hung Liu, Xuchen Song**[†], **Yang Li, Yahui Zhou**
Kunlun Inc.
{maxwy.lam, xuchen.song}@kunlun-inc.com

Abstract

Autoregressive (AR) models have demonstrated impressive capabilities in generating high-fidelity music. However, the conventional next-token prediction approach paradigm in AR models does not align with the human creative process in music composition, potentially compromising the musicality of generated samples. To overcome this limitation, we introduce MusiCoT, a groundbreaking chain-of-thought (CoT) prompting technique tailored for music generation. MusiCoT empowers the AR model to first outline an overall music structure before generating audio tokens, thereby enhancing the coherence and creativity of the resulting compositions. By leveraging the contrastive language-audio pretraining (CLAP) model, we establish a chain of "musical thoughts", making MusiCoT scalable and independent of human-labeled data, in contrast to conventional CoT methods. Moreover, MusiCoT allows for in-depth analysis of music structure, such as instrumental arrangements, and supports music referencing – accepting variable-length audio inputs as optional style references. This innovative approach effectively addresses copying issues, positioning MusiCoT as a vital practical method for music prompting. Our experimental results indicate that MusiCoT consistently achieves superior performance across both objective and subjective metrics, producing music quality that competes with state-of-the-art generation models.
Our samples are available at https://MusiCoT.github.io/.

1 Introduction

In recent years, the field of audio generation has experienced significant advancements [1–14] with the emergence of deep generative methods. Despite these developments, the challenge of producing high-fidelity and realistic music remains a formidable task. Music generation requires a delicate balance: integrating vocals with a rich tapestry of instruments while maintaining a coherent melodic and harmonic structure, all while ensuring the accuracy of the linguistic content. Human listeners are particularly attuned to musical dissonance, leaving little room for error in generated compositions. Additionally, creating realistic music demands that models adeptly capture the intricacies of the full frequency spectrum. This challenge is further compounded in long-context music generation, where achieving an optimal balance between generation quality and computational efficiency is crucial.

In the realm of music generation, three primary classes of models have emerged as dominant players: (1) autoregressive (AR) models [2–4, 15], (2) diffusion models [5, 7–12, 15], and (3) a hybrid approach that combines language models (LM) with diffusion models [12–14], referred to as the

[*] Equal contribution
[†] Corresponding author

Preprint. Under review.

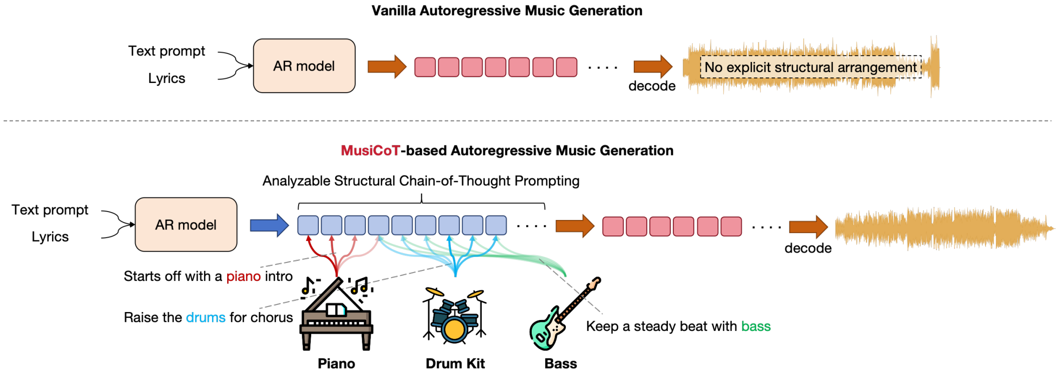
Vanilla Autoregressive Music Generation
Text prompt
Lyrics
AR model
decode
No explicit structural arrangement
MusiCoT-based Autoregressive Music Generation
Analyzable Structural Chain-of-Thought Prompting
Text prompt
Lyrics
AR model
decode
Starts off with a piano intro
Raise the drums for chorus
Keep a steady beat with bass
Piano
Drum Kit
Bass

Figure 1: This illustration showcases the MusiCoT reasoning process for music generation, focusing on instrumental arrangement. The arrows are color-coded to indicate the intensity of each instrument: darker colors represent higher intensity, while lighter shades signify lower intensity.

MeLoDy framework as in [12]. Each of these approaches has its own strengths and weaknesses, which are explored in detail in Section 2. This paper focuses on the promising MeLoDy framework, which serves as a backbone for this work. Within this framework, the language model plays a crucial role in aligning compositional inputs – such as text prompts and lyrics – with the generated musical content. However, it is important to note that this model operates under the paradigm of next-token prediction, which presents a significant limitation. Unlike the autoregressive nature of the model, the creative process of human music composition is inherently non-autoregressive [16]. Music producers typically engage in a thoughtful exploration of various elements – such as mood, genre, instruments, and melodies – before finalizing a music piece. Notably, this intermediate reasoning aligns well with the concept of chain-of-thought (CoT) prompting [17], where the LM is trained to generate solutions in a step-by-step manner using natural language. Despite the potential of applying CoT techniques to music AR models, this approach remains largely underexplored in the field of music generation.

In pursuit of high-fidelity music generation, this paper introduces MusiCoT, an innovative chain-of-thought (CoT) prompting technique specifically designed for music creation. As illustrated in Figure 1, MusiCoT enables the AR model to first establish a comprehensive and analyzable music structure before generating audio tokens. By leveraging the contrastive language-audio pretraining (CLAP) model [18], we define a coherent chain of "musical thoughts". The key contributions of MusiCoT are encapsulated in the 4S framework:

- **Scalability**: MusiCoT is built on a separately pretrained CLAP model, allowing for easy scaling with the base AR model without the need for human-labeled data.
- **Structural Analyzability**: In the light of CLAP, MusiCoT offers structural analyzability, facilitating the analysis of elements, e.g., instrumental arrangements, as shown in Figure 1.
- **Support for Music Reference**: By making slight adjustments to the inference strategy, MusiCoT seamlessly supports music referencing, enabling the input of variable-length audio as an optional style reference. Our experiments demonstrate that MusiCoT effectively mitigates copying issues, making it advantageous for abstractive music referencing.
- **Superior Generation Performance**: Empirical evidence shows that integrating MusiCoT within the MeLoDy framework consistently yields exceptional generation performance, as measured by both objective metrics and subjective evaluations, resulting in music quality that competes with state-of-the-art (SoTA) music generation models.

2 Related Work

2.1 Music Generation

In this section, we provide a concise overview of conventional music generation models, highlighting the significant advancements brought about by large language models (LLMs) [19–21] in reasoning capabilities. This has led to the emergence of autoregressive (AR) music generative models [2–4, 15], which have achieved groundbreaking results in the field. However, the performance of these AR-based methods is often limited by the VQ-VAE [1], which reconstructs waveforms from a sequence of codes but suffers from low bitrate constraints when utilizing a single codebook [3]. To address this limitation, recent works have introduced residual vector quantization (RVQ) techniques [22, 23], enhancing bitrate and promoting the use of multiple codebooks. Nonetheless, the complexity of AR modeling increases significantly when predicting tokens across multiple codebooks [2, 3].

On the other hand, diffusion-based music generative models [5, 7–12, 15], while operating in continuous space [24, 25] without the drawbacks of quantization loss, demonstrate the potential for superior audio generation quality. However, these models face challenges in supporting long-context windows due to the high computational costs associated with continuous latent vectors [25], which may compromise the musicality and structural coherence of generated samples.

Recognizing the limitations of both approaches, researchers have introduced a cascade model of LMs and diffusion, known as MeLoDy [12]. This framework leverages an LM to predict semantic tokens from a single codebook derived from self-supervised learning (SSL). These tokens then guide a diffusion model for fine-grained waveform generation. As a result, MeLoDy not only produces high-quality music audio comparable to diffusion-based methods but also accommodates long-context windows akin to AR models. This promising cascade approach is rapidly gaining traction in both music generation [13, 14] and speech synthesis [26, 27], showcasing its transformative potential.

2.2 Chain-of-Thought Prompting

As a groundbreaking work, Wei et al. [17] introduced the concept of chain-of-thought (CoT) prompting, which involves generating a sequence of intermediate reasoning steps – akin to a chain of thoughts – within large language models (LLMs). Their findings indicate that CoT significantly enhances the reasoning capabilities of these models, sparking a wave of subsequent research [28–34]. CoT has also proven instrumental in recent industrial advancements, such as OpenAI's O1 [35] and DeepSeek-R1 [21], showcasing its practical value in language modeling.

However, training CoT models to articulate intermediate reasoning processes in natural language can be prohibitively expensive, particularly in the context of music generation.[1] Fortunately, Hao et al. [34] revealed that reasoning within a latent space offers LLMs the flexibility to think without the constraints of language. This notion is supported by neuroimaging research [36], which suggests that human language is primarily designed for communication rather than reasoning. This insight aligns with the philosophy of our proposed MusiCoT framework, which similarly emphasizes an abstractive latent space over costly traditional natural-language-based CoT approaches.

Recent studies in the audio domain have also explored the CoT technique [15, 37, 38]. Notably, YuE [15] presents a CoT method for music generation that focuses on music segments (e.g. verse, chorus, bridge, etc.) [39] extracted by the All-in-one tool [40]. However, this approach differs fundamentally from our MusiCoT framework, which offers a more nuanced perspective on music generation.

3 Music Generation Framework

This section introduces the foundational music generation framework for our model, specifically the **MeLoDy** (**M** for music; **L** for LM; **D** for diffusion). MeLoDy [12] serves as an efficient alternative to MusicLM [2], giving impressive results in music generation. The MeLoDy framework comprises two distinct modeling stages: the semantic stage and the acoustic stage, which are detailed below.

3.1 Semantic Modeling with a Semantic Language Model

In this paper, we define the set of conditional inputs for music generation as $\mathbf{C}$. Our focus is on generating music with and without vocals, represented by $\mathbf{C} := \{\mathbf{c}_{\text{lyrics}}, \mathbf{c}_{\text{text}}\}$. Here, $\mathbf{c}_{\text{lyrics}}$ specifies the lyrics for the vocal parts, while $\mathbf{c}_{\text{text}}$ serves as a prompt describing the desired characteristics of the music. We extend the original MeLoDy framework, which was limited to non-vocal text-to-music generation, allowing for the optional omission of $\mathbf{c}_{\text{lyrics}}$ in instrumental music generation.

[1] Manually transcribing the intermediate reasoning involved in the music creative process for CoT prompting can be prohibitively costly, as it necessitates the expertise of qualified music professionals.

With this definition of $\mathbf{C}$, a semantic LM, parameterized by $\boldsymbol{\theta}$, addresses the following problem:

$$p_{\boldsymbol{\theta}}(\mathbf{s}_{1:N}|\mathbf{C}) = p_{\boldsymbol{\theta}}(\mathbf{s}_1|\mathbf{C}) \prod_{n=2}^{N} p_{\boldsymbol{\theta}}(\mathbf{s}_n|\mathbf{s}_{1:n-1}, \mathbf{C}), \tag{1}$$

where $\mathbf{s}_{1:N} = [\mathbf{s}_1, \ldots, \mathbf{s}_n]$ represents N semantic tokens generated through quantization methods like K-Means [41] or VQ-VAE [42], using self-supervised learning (SSL) encoders such as w2v-BERT [2, 43], Wav2Vec2-Conformer [12, 44, 45], BEST-RQ [14, 46], or fused X-Codec [15, 47]. The goal of semantic modeling is to capture meaningful representations, including phonemes, melodies, genres, and instruments. Although the MeLoDy framework accommodates various audio tokenizers, the choice of semantic tokenizer is crucial for the performance of the semantic LM and the quality of the final audio. In this study, we adopt the BEST-RQ as our SSL encoder, as detailed in Section 5.

3.2 Acoustic Modeling with a Diffusion Model

Unlike MusicLM [2], which utilizes two distinct LMs for coarse and fine acoustic modeling through multi-codebook prediction, MeLoDy offers a more streamlined and effective solution by replacing these LMs with a single diffusion model. This diffusion model, parameterized by $\boldsymbol{\phi}$, is designed to recover the true music data distribution $p_{\text{data}}(\mathbf{x})$ using a variational distribution $p_{\boldsymbol{\phi}}(\mathbf{x}_0|\mathbf{s}_{1:N})$. It achieves this by executing T steps of reverse process, starting from a prior $p(\mathbf{x}_T) := \mathcal{N}(\mathbf{0}, \mathbf{I})$:

$$p_{\boldsymbol{\phi}}(\mathbf{x}_0|\mathbf{s}_{1:N}) = \mathbb{E}_{\mathbf{x}_1,\ldots,\mathbf{x}_T}\left[p(\mathbf{x}_T)\prod_{t=1}^{T} p_{\boldsymbol{\phi}}(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{s}_{1:N})\right]. \tag{2}$$

Additionally, the authors of MeLoDy [12] introduce a novel dual-path diffusion (DPD) based on an innovative network architecture for the v-diffusion model [48]. In this paper, we generalize this approach, allowing for flexibility in the diffusion model's architecture, training algorithms, and sampling methods. After reviewing several successful diffusion models in the audio domain, we have chosen the diffusion model presented in Stable Audio [11], which is elaborated upon in Section 5.

3.3 Conditional Music Generation with Lyrics and Text Prompts

Using text prompts to guide generative models is a common approach in music generation. Previous studies [2, 3, 9, 11, 12] typically encode these prompts, denoted as $\mathbf{c}_{\text{text}}$, into continuous embeddings via a text encoder. However, accurately capturing the intent of these descriptions can be challenging. In our framework, we ensure that the text embedding aligns with the generated semantic tokens by prefixing it to the semantic LM. Notably, MusicLM [2] and MeLoDy [12] leverage the MuLan model [49], which jointly embeds music audio and its corresponding text into a joint embedding space, minimizing divergence between them. Similarly, the contrastive language-audio pretraining (CLAP) model [18] is widely used in text-to-music generation, e.g. in [3, 9, 11], serving as an alternative to MuLan. During training, we prefix the audio embeddings to the lyrics, and at inference, we can seamlessly switch to using text embeddings for music generation. For optimal performance, we have developed our own CLAP model, which will be detailed in the Appendix.

In addition to cross-domain embedding models, we can utilize outputs from music information retrieval (MIR) models [50], which categorize music into tags such as genres, genders, moods, and instruments. This MIR-based prompting method, also employed in YuE [15], is considered an in-context learning (ICL) approach for music generation. Our findings indicate that combining CLAP-based and MIR-based prompting yields the highest accuracy and quality in generated music samples. Then, the structure of conditions fed into the semantic LM can be formally written as

$$\mathbf{c}_{\text{clap}} \oplus \text{Emb}_{\boldsymbol{\theta}}\left(\text{Tokenizer}(\mathbf{c}_{\text{tags}})\right) \oplus \text{Emb}_{\boldsymbol{\theta}}\left(\text{Tokenizer}(\mathbf{c}_{\text{lyrics}})\right), \tag{3}$$

where $\oplus$ signifies concatenation along the feature dimension. Here, $\text{Emb}_{\boldsymbol{\theta}}(\cdot)$ refers to the token embedder trained in conjunction with the semantic LM, and $\text{Tokenizer}(\cdot)$ is the method employed for natural language tokenization, such as the subword-based approach using the BERT tokenizer [51].

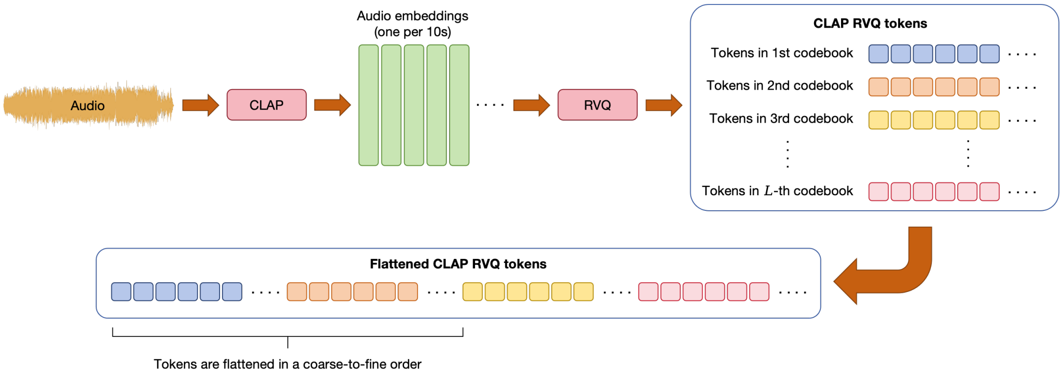
Audio
CLAP
Audio embeddings (one per 10s)
RVQ
CLAP RVQ tokens
Tokens in 1st codebook
Tokens in 2nd codebook
Tokens in 3rd codebook
Tokens in L-th codebook
Flattened CLAP RVQ tokens
Tokens are flattened in a coarse-to-fine order

Figure 2: The diagram illustrating the computation of flattened CLAP RVQ tokens given an audio.

In addition, we define the components as follows:

$$\mathbf{c}_{\text{clap}} = \begin{cases} \text{CLAP}_{\text{audio}}(\mathbf{x}), & \text{at training time;} \\ \text{CLAP}_{\text{text}}(\mathbf{c}_{\text{text}}), & \text{at inference time,} \end{cases} \tag{4}$$

$$\mathbf{c}_{\text{tags}} = \begin{cases} \text{MIR}(\mathbf{x}), & \text{at training time;} \\ \left\{ \forall \mathbf{c}_{\text{tag}} \in \mathbb{T} \mid \frac{\text{CLAP}_{\text{text}}(\mathbf{c}_{\text{text}})^\top \text{CLAP}_{\text{text}}(\mathbf{c}_{\text{tag}})}{\|\text{CLAP}_{\text{text}}(\mathbf{c}_{\text{text}})\|_2^2 \|\text{CLAP}_{\text{text}}(\mathbf{c}_{\text{tag}})\|_2^2} > \delta \right\}, & \text{at inference time,} \end{cases} \tag{5}$$

where $\mathbf{x} \sim p_{\text{data}}(\mathbf{x})$ represents the training music audio, $\text{CLAP}_{\text{audio}}(\cdot)$ and $\text{CLAP}_{\text{text}}(\cdot)$ denote the audio[2] and text encoders in the CLAP model, respectively. The function $\text{MIR}(\cdot)$ encompasses various music information retrieval classifiers, including those for genre, gender, mood, and instrument. The set $\mathbb{T}$ includes all possible outputs from the $\text{MIR}(\cdot)$ classifiers, and $\delta \in (0, 1)$ serves as a threshold for the cosine similarity between the text prompt and each tag in the set.

4 MusiCoT: Analyzable Chain-of-Musical-Thought Prompting

As highlighted in [52], "musical thought" is the foundation of a music producer's creativity. When composing or improvising, producers engage in a distinct decision-making process, effectively "thinking musically". This creative journey often involves breaking down the process into intermediate decisions, refining each choice before finalizing the piece. A primary goal of this paper is to equip music generative models with the capability to replicate this chain of musical thought – creating a coherent series of reasoning and decision-making steps that culminate in a polished music sample.

4.1 Viewing CLAP Audio Embeddings as Analyzable Musical Thoughts

This paper proposes a novel approach to representing intermediate musical thoughts using the contrastively trained cross-domain embedding model, known as the CLAP model [18], rather than relying on natural language descriptions as seen in [17]. The concept of utilizing continuous features is not new; previous research by Hao et al. [34] indicates that reasoning in a latent space is often more effective than reasoning in natural language. This is further supported by neuro-imaging studies [36], which suggest that human language is primarily optimized for communication rather than for reasoning tasks. Specifically, the CLAP model encodes segments of music audio into continuous-valued embeddings every 10 seconds. For a typical 3-minute song, this results in a sequence of audio embeddings, denoted as $\mathbf{C}_{\text{clap}} := \left[\mathbf{c}_{\text{clap}}^{(1)}, \ldots, \mathbf{c}_{\text{clap}}^{(M)}\right]$. Each embedding, corresponding to a 10-second clip, is analyzable that allows for cosine similarity calculations against any relevant text.

4.2 Predicting Coarse-to-Fine Flattened RVQ for a Stabler MusiCoT Training

With the chain of musical thoughts established, we encounter a significant challenge. The continuous nature of CLAP audio embeddings renders traditional training objectives, such as mean-squared-

[2] Since the audio encoder convert every 10s of audio into an embedding, we randomly choose one of embeddings at training time to improve robustness.

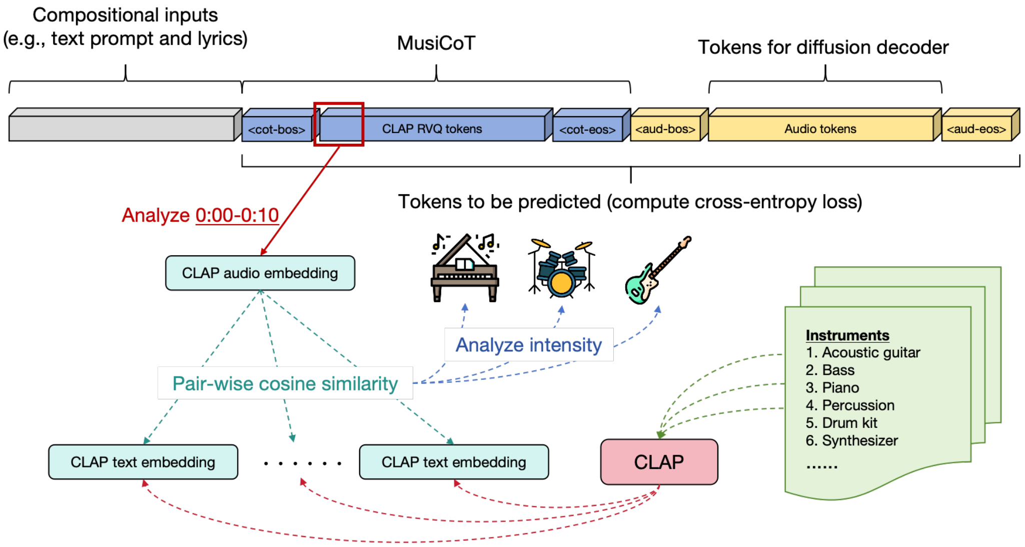

Figure 3: The diagram presenting the token arrangement in MusiCoT-based autoregressive model and the structural analyzability obtained from the CLAP RVQ token prediction.

error (MSE) loss, L1 loss, and contrastive infoNCE loss [53], ineffective for music generation. The aforementioned work Hao et al. [34] has not explicitly addressed the prediction of continuous thoughts, depends instead on standard CoT training in natural language.

To tackle this issue in MusiCoT, we introduce a residual vector quantization (RVQ) [22] based coarse-to-fine tokenization method, illustrated in Figure 2. This RVQ model consists of L codebooks, parameterized by $\boldsymbol{\zeta}$, and trained using a reconstruction-based quantization loss. Each frame of audio embedding, denoted as $\mathbf{c}_{\text{clap}}^{(m)}$, is discretized into tokens $c_{\text{clap}}^{(m,1)}, \ldots, c_{\text{clap}}^{(m,L)}$, leading to a quantized vector $\tilde{\mathbf{c}}_{\text{clap}}^{(m)} := \tilde{\mathbf{c}}_{\text{clap}}^{(m,L)}$ as defined below:

$$\tilde{\mathbf{c}}_{\text{clap}}^{(m,k)} = \sum_{k=1}^{k} \text{Emb}_{\boldsymbol{\zeta}}^{(k)} \left(c_{\text{clap}}^{(m,k)} \right), \quad c_{\text{clap}}^{(m,k)} = \operatorname*{arg\,min}_{q \in \mathbb{Q}^{(k)}} \left\| \tilde{\mathbf{c}}_{\text{clap}}^{(m,k)} - \text{Emb}_{\boldsymbol{\zeta}}^{(k)}(q) \right\|_2^2 \tag{6}$$

where $\text{Emb}_{\boldsymbol{\zeta}}^{(k)}(\cdot)$ is the k-th residual token embedder for the k-th codebook, $\tilde{\mathbf{c}}_{\text{clap}}^{(m,k)}$ denotes the cumulative sum of the first k quantized vectors, and $\mathbb{Q}^{(k)}$ is the index set of the k-th codebook.

In MusiCoT, we arrange the RVQ tokens in a flattened coarse-to-fine sequence for LM prediction, ensuring that coarser tokens are predicted before finer ones. Unlike traditional CoT reasoning, which breaks down complex tasks into smaller steps, music generation requires a holistic approach. Our intermediate musical thoughts are designed to maintain this integrity, with each token sequence corresponding directly to the whole generated music with precise time alignment. The L codebooks represent different levels of granularity, making the generation of these intermediate tokens akin to structuring music from a broad to detailed perspective.

During training, the semantic LM utilizes the flattened CLAP RVQ tokens as additional prediction targets, as shown in Figure 3. Similar to standard CoT training, these predicted tokens are treated like audio tokens for cross-entropy (CE) loss computation, with the addition of two special tokens – <cot_bos> and <cot_eos> – to indicate when to transition from generating MusiCoT tokens to audio tokens. The inherent structure of CLAP embeddings allows for the analysis of predicted RVQ tokens within a joint language-audio latent space, enabling us to explore the musical characteristics of each 10-second audio segment. For instance, we can analyze the arrangement of instruments by computing cosine similarities between generated embeddings and the text embeddings of various instruments, providing insights into how different instruments interact over time in the generated music.

4.3 Dual-Sampling Strategy for MusiCoT

In MusiCoT, we integrate tokens from three domains: text tokens, flattened CLAP RVQ tokens, and audio tokens, into a single LM. This raises an important question regarding the sampling strategy: should we use the same sampling approach for both the flattened CLAP RVQ tokens and the audio tokens, or should we adopt different strategies? This issue is relatively unexplored in existing literature. In this section, we present two novel dual-sampling strategies specifically designed for MusiCoT.

4.3.1 Dual-Temperature Sampling

A recent study [54] highlights the critical role of temperature selection as a sampling hyperparameter in enhancing language model (LM) performance. Our experimental findings in music generation further support this importance. To leverage this insight, we introduce a dual-temperature sampling method for MusiCoT. This approach involves configuring the semantic LM with two distinct sets of sampling temperatures: one for the flattened CLAP RVQ tokens and another for the audio tokens. The effectiveness of this dual-temperature sampling strategy is demonstrated in Section 5.

4.3.2 Dual-Scale Classifier-Free Guidance

Classifier-free guidance (CFG) [55] is a versatile technique originally developed for diffusion generative models. Its effectiveness has also been demonstrated in language modeling applications, including AudioGen [56] and MusicGen [3]. In our research, we have identified significant benefits of employing CFG within both the semantic language model and the diffusion model, even though the authors of MeLoDy [12] did not explore CFG-based sampling for the semantic LM. For MusiCoT, we introduce a dual-scale CFG sampling strategy that modifies the log probabilities as follows:

$$\log p_{\boldsymbol{\theta}}(\mathbf{c}_{\text{clap}}^{(1:M,1:L)}|\mathbf{C}) = \lambda_1 \log p_{\boldsymbol{\theta}}(\mathbf{c}_{\text{clap}}^{(1:M,1:L)}|\mathbf{C}) + (1-\lambda_1) \log p_{\boldsymbol{\theta}}(\mathbf{c}_{\text{clap}}^{(1:M,1:L)}), \tag{7}$$

$$\log p_{\boldsymbol{\theta}}(\mathbf{s}_{1:N}|\mathbf{c}_{\text{clap}}^{(1:M,1:L)}, \mathbf{C}) = \lambda_2 \log p_{\boldsymbol{\theta}}(\mathbf{s}_{1:N}|\mathbf{c}_{\text{clap}}^{(1:M,1:L)}, \mathbf{C}) + (1-\lambda_2) \log p_{\boldsymbol{\theta}}(\mathbf{s}_{1:N}), \tag{8}$$

where $\mathbf{c}_{\text{clap}}^{(1:M,1:L)} := [c_{\text{clap}}^{(1,1)}, \ldots, c_{\text{clap}}^{(M,1)}, c_{\text{clap}}^{(1,2)}, \ldots, c_{\text{clap}}^{(M,2)}, \ldots, c_{\text{clap}}^{(M,L)}]$ represents the flattened CLAP RVQ tokens.

5 Experiments

5.1 Experimental Setup

Model Setup In this work, we employ the LLaMA architecture [57], as suggested in [12], for the semantic LM, except opting for a larger variant with approximately 1 billion parameters. For the self-supervised learning (SSL) model that generates audio tokens, we utilize the BEST-RQ model [46] with a frame rate of 25Hz. For better pronunciation clarity, the BEST-RQ model is fine-tuned with CTC loss [58] given paired music audio and lyrics, in a way similar to the approach in [59]. Our diffusion model mirrors the architecture, training and sampling pipeline of Stable Audio[3] [60], also featuring a model size of about 1 billion parameters to convert audio tokens into high-quality waveforms. Additionally, we replicate the audio VAE-GAN used in Stable Audio, reconstructing audio at 44.1kHz with 43Hz 64-dim latents. For the CLAP model, we train a music-focused version using the official implementation[4]. Following this, we train the RVQ model using a publicly available implementation[5]. Detailed configurations can be found in the Appendix.

Data Preparation The models in this paper – including SSL, CLAP, RVQ, semantic LM, audio VAE-GAN, and diffusion model – are trained on approximately 10 million English songs sourced from DISCO-10M [61] and around 200,000 confidential in-house music tracks. Data preprocessing is essential to meet the diverse requirements of each model. Initially, we use the Demucs [62–64] music source separation model to extract vocals from the songs. Next, an automatic speech recognition (ASR) model transcribes the extracted vocals and provides timestamps for each line of lyrics. We

[3] https://github.com/Stability-AI/stable-audio-tools

[4] https://github.com/LAION-AI/CLAP

[5] https://github.com/lucidrains/vector-quantize-pytorch

Table 1: We compare our base model, MusiCoT, with conventional music generation models, including leading commercial products.[6] The asterisk (*) indicates our own implementation of the base model within the MeLoDy framework for music generation.

Music Generator	**RTF** (↓)	**MOS** (↑)	**FAD** (↓)	**Content Scores** (↑)			
				CE	**CU**	**PC**	**PQ**
Suno V4 [66]	0.84	**3.77**	0.122	**7.58**	**7.87**	6.28	**8.21**
Udio V1.5 [67]	1.48	3.62	**0.110**	7.14	7.29	**6.72**	7.46
Mureka V5.5 [68]	**0.27**	3.39	0.119	7.34	7.76	6.41	8.03
YuE[7] [15]	12	3.00	0.287	7.12	7.51	5.30	7.78
*MeLoDy [12]	**0.27**	3.35	0.112	7.49	7.85	**6.38**	**8.11**
*MeLoDy [12] + **MusiCoT**	**0.27**	**3.72**	**0.102**	**7.49**	**7.87**	6.21	**8.11**
w/o Dual-Temp.	**0.27**	3.49	0.111	7.46	7.83	6.18	8.06
w/o DS-CFG	**0.27**	3.51	0.106	7.47	7.84	6.09	8.06
w/o Dual-Temp. & DS-CFG	**0.27**	3.48	0.113	7.41	7.82	6.2	8.03

also employ a voice activity detection (VAD) model to identify silent segments. To segment the music into parts like intro, verse, chorus, break, and outro, we utilize the All-in-One model [40]. For evaluation, we generate 100 sets of English lyrics using ChatGPT [65], incorporating various moods and imaginative scenarios. Additionally, we use ChatGPT to create tags that describe genre, sub-genres, gender, mood, instruments, and subjective feelings. To ensure fairness in comparisons, all models are tested with the same pairs of lyrics and prompts.

Evaluation Metrics To demonstrate the efficiency of the MeLoDy framework [12], we evaluate the real-time factor (RTF) – the time taken to generate one second of audio – across various commercial products and models running on a NVIDIA RTX 4090 GPU. For subjective assessment, we enlist ten music professionals to rate the overall quality of the generated music on a scale from 1 to 5, focusing on fidelity, musicality, and creativity. The mean opinion score (MOS) is reported for comparative analysis, with detailed evaluation protocols provided in the Appendix. For objective metrics, we utilize the CLAP-based Fréchet audio distance (FAD) [69] to measure the fidelity of the generated audio against reference tracks from MUSDB18-HQ [70]. It is important to note that while the MusicCaps [2] test set is commonly used for FAD calculations, the audio quality in MusicCaps, sourced from the Audio Set [71], is significantly inferior to commercially produced music, leading to inaccurate fidelity comparisons – a concern echoed by the music professionals. Furthermore, previous studies often report VGGish-based FAD, e.g., in [2, 3, 9, 15]; however, this approach requires downsampling audio to mono 16kHz, which compromises sound quality and renders it less sensitive to quality differences. In contrast, CLAP supports stereo audio at 48kHz, making it more suitable for quality comparisons. Additionally, we employ the Meta Audiobox-Aesthetic [72], which leverages advanced neural networks to assess perceived musical aesthetics, including content enjoyment (CE), content usefulness (CU), production complexity (PC), and production quality (PQ).

5.2 Comparing with the State-of-The-Art Models

Table 1 summarizes the results of our comparisons. We first assessed the performance of MusiCoT by contrasting the MeLoDy base model with and without MusiCoT. Notably, incorporating MusiCoT does not increase inference time, as measured by the RTF, yet it significantly boosts the MOS and slightly improves the FAD and content scores. This confirms that MusiCoT effectively enhances music generation performance. We then benchmarked our model against three leading closed-source music generation products: Suno V4 [66], Udio V1.5 [67], and Mureka V5.5 [68]. It is important to note that our evaluation, conducted in March 2025, reflects the performance of these products at that specific time due to their black-box nature. As shown in Table 1, while Mureka V5.5 excels in generation speed, its quality lags behind Suno V4 and Udio V1.5. Suno V4 achieved the highest MOS, indicating superior overall music quality, whereas Udio V1.5 recorded the lowest FAD and

[6] Notably, previous works in academic research have struggled to match the performance of these commercial offerings, often omitting them due to significant performance gaps.

[7] YuE is tested using the model released at https://github.com/multimodal-art-projection/YuE

higher production complexity (PC), attributed to its exceptional sound quality. Before introducing MusiCoT, our MeLoDy base model lagged behind all commercial products. However, with MusiCoT, we significantly elevate our model's MOS, positioning it as the second-best among all competitors. Furthermore, our MusiCoT-based model achieves the lowest FAD among all competitors, highlighting the high fidelity of the generated music and underscoring the practical value of MusiCoT. We invite you to visit our demo page[8] to experience the enhanced fidelity brought by MusiCoT.

5.3 Structural Analyzability with MusiCoT

A key feature of MusiCoT is its capability to analyze intermediate musical content using predicted CLAP RVQ tokens in conjunction with various text anchors. These text anchors are a fixed set of embeddings that represent common music-related terms, such as instrument names. To illustrate this functionality, we utilized the 'htdemucs_6s' model from Demucs [62–64], which separates audio into six distinct tracks: vocals, bass, drums, piano, guitar, and other accompaniment. We selected five text anchors –'vocals', 'bass', 'drums', 'guitar', and 'piano' – each defined by unique musical characteristics. By calculating the cosine similarities between the quantized CLAP embeddings derived from the generated RVQ tokens and these text anchors, we assessed the correlation with the corresponding track volumes using the Pearson correlation coefficient (r). Over 30-second segments, we found the average correlation coefficients for each text anchor: $r_{\text{vocals}} = 0.689$, $r_{\text{bass}} = 0.584$, $r_{\text{drums}} = 0.639$, $r_{\text{guitar}} = 0.628$, and $r_{\text{piano}} = 0.531$. To neglect noise from the music separator, we excluded any track with an average volume below 10^{-2} from our calculations. Notably, these results reveal a positive correlation between the cosine similarities and track volumes, reinforcing our hypothesis regarding the analyzability of MusiCoT.

5.4 Music Referencing with MusiCoT

One of the standout features of MusiCoT is its seamless support for music referencing. By extracting CLAP RVQ tokens from a reference track, MusiCoT can efficiently transition to predicting audio tokens. This capability enhances music generation by allowing variable-length audio inputs, unlike traditional CLAP-based methods that are limited to fixed lengths, as noted in [12]. Additionally, while continuation-based strategies (such as ICL [15]) offer another approach to audio referencing, they carry a significant risk of the model replicating token sequences from training data. MusiCoT addresses this concern effectively, as CLAP RVQ tokens are more abstract and less prone to copying than standard audio tokens. For practical examples of music referencing, please visit our demo page.

5.5 Ablation study on Dual-Sampling Strategies

In this section, we explore the impact of two dual-sampling strategies: dual-temperature sampling (Dual-Temp.) and dual-scale classifier-free guidance (DS-CFG), as introduced in Section 4.3. For Dual-Temp., we set the temperature for MusiCoT tokens at 0.65 and for audio tokens at 0.75. In the case of DS-CFG, we use $\lambda_1 = 2.3, \lambda_2 = 1.3$, determined through a grid search algorithm. Our findings, presented in lower part of Table 1, indicate that omitting either or both strategies results in a decline across all performance metrics. Notably, the combined application of Dual-Temp. and DS-CFG yields a more substantial enhancement in performance than utilizing either strategy alone.

6 Conclusion

In conclusion, this paper presents MusiCoT, a novel chain-of-thought prompting technique that enhances high-fidelity music generation by aligning the creative processes of AR models with musical thought. By leveraging the CLAP model, MusiCoT not only ensures structural analyzability but also supports music referencing, thereby elevating the quality of generated music. Our experimental results demonstrate that MusiCoT consistently leads to superior generation performances, establishing itself as a valuable advancement in music generation. Ultimately, MusiCoT paves the way for future explorations of generative AI, merging artificial intelligence with the artistry of human creativity.

[8] https://MusiCoT.github.io

References

[1] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*, 2020.

[2] Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, et al. Musiclm: Generating music from text. *arXiv preprint arXiv:2301.11325*, 2023.

[3] Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. Simple and controllable music generation. *Advances in Neural Information Processing Systems*, 36:47704–47720, 2023.

[4] Julian D Parker, Janne Spijkervet, Katerina Kosta, Furkan Yesiler, Boris Kuznetsov, Ju-Chiang Wang, Matt Avent, Jitong Chen, and Duc Le. Stemgen: A music generation model that listens. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1116–1120. IEEE, 2024.

[5] Marco Pasini and Jan Schlüter. Musika! fast infinite waveform music generation. *arXiv preprint arXiv:2208.08706*, 2022.

[6] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. *arXiv preprint arXiv:2301.12503*, 2023.

[7] Flavio Schneider, Zhijing Jin, and Bernhard Schölkopf. Moûsai: Text-to-music generation with long-context latent diffusion. *arXiv preprint arXiv:2301.11757*, 2023.

[8] Qingqing Huang, Daniel S Park, Tao Wang, Timo I Denk, Andy Ly, Nanxin Chen, Zhengdong Zhang, Zhishuai Zhang, Jiahui Yu, Christian Frank, et al. Noise2music: Text-conditioned music generation with diffusion models. *arXiv preprint arXiv:2302.03917*, 2023.

[9] 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. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1206–1210. IEEE, 2024.

[10] Peike Patrick Li, Boyu Chen, Yao Yao, Yikai Wang, Allen Wang, and Alex Wang. Jen-1: Text-guided universal music generation with omnidirectional diffusion models. In *2024 IEEE Conference on Artificial Intelligence (CAI)*, pages 762–769. IEEE, 2024.

[11] Zach Evans, CJ Carr, Josiah Taylor, Scott H Hawley, and Jordi Pons. Fast timing-conditioned latent audio diffusion. In *Forty-first International Conference on Machine Learning*, 2024.

[12] Max WY Lam, Qiao Tian, Tang Li, Zongyu Yin, Siyuan Feng, Ming Tu, Yuliang Ji, Rui Xia, Mingbo Ma, Xuchen Song, et al. Efficient neural music generation. *Advances in Neural Information Processing Systems*, 36:17450–17463, 2023.

[13] Ye Bai, Haonan Chen, Jitong Chen, Zhuo Chen, Yi Deng, Xiaohong Dong, Lamtharn Hantrakul, Weituo Hao, Qingqing Huang, Zhongyi Huang, et al. Seed-music: A unified framework for high quality and controlled music generation. *arXiv preprint arXiv:2409.09214*, 2024.

[14] Shun Lei, Yixuan Zhou, Boshi Tang, Max WY Lam, Hangyu Liu, Jingcheng Wu, Shiyin Kang, Zhiyong Wu, Helen Meng, et al. Songcreator: Lyrics-based universal song generation. *Advances in Neural Information Processing Systems*, 37:80107–80140, 2024.

[15] Ruibin Yuan, Hanfeng Lin, Shuyue Guo, Ge Zhang, Jiahao Pan, Yongyi Zang, Haohe Liu, Yiming Liang, Wenye Ma, Xingjian Du, et al. Yue: Scaling open foundation models for long-form music generation. *arXiv preprint arXiv:2503.08638*, 2025.

[16] Richard James Burgess. *The art of music production: The theory and practice*. Oxford University Press, 2013.

[17] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[18] Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. Clap learning audio concepts from natural language supervision. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[19] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[20] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[21] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[22] 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.

[23] Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-fidelity audio compression with improved rvqgan. *Advances in Neural Information Processing Systems*, 36:27980–27993, 2023.

[24] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[25] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[26] Philip Anastassiou, Jiawei Chen, Jitong Chen, Yuanzhe Chen, Zhuo Chen, Ziyi Chen, Jian Cong, Lelai Deng, Chuang Ding, Lu Gao, et al. Seed-tts: A family of high-quality versatile speech generation models. *arXiv preprint arXiv:2406.02430*, 2024.

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

[28] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

[29] Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*, 2022.

[30] Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. Faithful chain-of-thought reasoning. In *The 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (IJCNLP-AACL 2023)*, 2023.

[31] Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. Multimodal chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*, 2023.

[32] Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36:74952–74965, 2023.

[33] Guhao Feng, Bohang Zhang, Yuntian Gu, Haotian Ye, Di He, and Liwei Wang. Towards revealing the mystery behind chain of thought: a theoretical perspective. *Advances in Neural Information Processing Systems*, 36:70757–70798, 2023.

[34] Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. *arXiv preprint arXiv:2412.06769*, 2024.

[35] Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv preprint arXiv:2412.16720*, 2024.

[36] Evelina Fedorenko, Steven T Piantadosi, and Edward AF Gibson. Language is primarily a tool for communication rather than thought. *Nature*, 630(8017):575–586, 2024.

[37] Yexing Du, Ziyang Ma, Yifan Yang, Keqi Deng, Xie Chen, Bo Yang, Yang Xiang, Ming Liu, and Bing Qin. Cot-st: Enhancing llm-based speech translation with multimodal chain-of-thought. *arXiv preprint arXiv:2409.19510*, 2024.

[38] Xinsheng Wang, Mingqi Jiang, Ziyang Ma, Ziyu Zhang, Songxiang Liu, Linqin Li, Zheng Liang, Qixi Zheng, Rui Wang, Xiaoqin Feng, et al. Spark-tts: An efficient llm-based text-to-speech model with single-stream decoupled speech tokens. *arXiv preprint arXiv:2503.01710*, 2025.

[39] Oriol Nieto, Gautham J Mysore, Cheng-i Wang, Jordan BL Smith, Jan Schlüter, Thomas Grill, and Brian McFee. Audio-based music structure analysis: Current trends, open challenges, and applications. *Transactions of the International Society for Music Information Retrieval*, 3(1), 2020.

[40] Taejun Kim and Juhan Nam. All-in-one metrical and functional structure analysis with neighborhood attentions on demixed audio. In *2023 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 1–5. IEEE, 2023.

[41] Mohiuddin Ahmed, Raihan Seraj, and Syed Mohammed Shamsul Islam. The k-means algorithm: A comprehensive survey and performance evaluation. *Electronics*, 9(8):1295, 2020.

[42] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[43] 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. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 244–250. IEEE, 2021.

[44] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460, 2020.

[45] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al. Conformer: Convolution-augmented transformer for speech recognition. *arXiv preprint arXiv:2005.08100*, 2020.

[46] Chung-Cheng Chiu, James Qin, Yu Zhang, Jiahui Yu, and Yonghui Wu. Self-supervised learning with random-projection quantizer for speech recognition. In *International Conference on Machine Learning*, pages 3915–3924. PMLR, 2022.

[47] Zhen Ye, Peiwen Sun, Jiahe Lei, Hongzhan Lin, Xu Tan, Zheqi Dai, Qiuqiang Kong, Jianyi Chen, Jiahao Pan, Qifeng Liu, et al. Codec does matter: Exploring the semantic shortcoming of codec for audio language model. *arXiv preprint arXiv:2408.17175*, 2024.

[48] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

[49] 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.

[50] Alexander Lerch. *An introduction to audio content analysis: Music Information Retrieval tasks and applications*. John Wiley & Sons, 2022.

[51] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[52] Christopher Bartel. Musical thought and compositionality. 2006.

[53] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[54] Weihua Du, Yiming Yang, and Sean Welleck. Optimizing temperature for language models with multi-sample inference. *arXiv preprint arXiv:2502.05234*, 2025.

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

[56] Felix Kreuk, Gabriel Synnaeve, Adam Polyak, Uriel Singer, Alexandre Défossez, Jade Copet, Devi Parikh, Yaniv Taigman, and Yossi Adi. Audiogen: Textually guided audio generation. *arXiv preprint arXiv:2209.15352*, 2022.

[57] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[58] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376, 2006.

[59] Yu Zhang, Wei Han, James Qin, Yongqiang Wang, Ankur Bapna, Zhehuai Chen, Nanxin Chen, Bo Li, Vera Axelrod, Gary Wang, et al. Google usm: Scaling automatic speech recognition beyond 100 languages. *arXiv preprint arXiv:2303.01037*, 2023.

[60] Zhifeng Kong and Wei Ping. On fast sampling of diffusion probabilistic models. In *ICML Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models*, 2021.

[61] Luca Lanzendörfer, Florian Grötschla, Emil Funke, and Roger Wattenhofer. Disco-10m: A large-scale music dataset. *Advances in Neural Information Processing Systems*, 36:54451–54471, 2023.

[62] Alexandre Défossez, Nicolas Usunier, Léon Bottou, and Francis Bach. Demucs: Deep extractor for music sources with extra unlabeled data remixed. *arXiv preprint arXiv:1909.01174*, 2019.

[63] Alexandre Défossez. Hybrid spectrogram and waveform source separation. In *Proceedings of the ISMIR 2021 Workshop on Music Source Separation*, 2021.

[64] Simon Rouard, Francisco Massa, and Alexandre Défossez. Hybrid transformers for music source separation. In *ICASSP 23*, 2023.

[65] OpenAI. Chatgpt. *URL https://chat.openai.com/*, 2023.

[66] Suno team. Introducing v4. *URL https://suno.com/blog/v4*, 2024.

[67] Udio team. Introducing v1.5. *URL https://www.udio.com/blog/introducing-v1-5*, 2024.

[68] Mureka team. Mureka ai. *URL https://www.mureka.ai*, 2024.

[69] Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi. Fréchet audio distance: A reference-free metric for evaluating music enhancement algorithms. In *INTERSPEECH*, pages 2350–2354, 2019.

[70] Zafar Rafii, Antoine Liutkus, Fabian-Robert Stöter, Stylianos Ioannis Mimilakis, and Rachel Bittner. Musdb18-hq-an uncompressed version of musdb18. *(No Title)*, 2019.

[71] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 776–780. IEEE, 2017.

[72] Andros Tjandra, Yi-Chiao Wu, Baishan Guo, John Hoffman, Brian Ellis, Apoorv Vyas, Bowen Shi, Sanyuan Chen, Matt Le, Nick Zacharov, et al. Meta audiobox aesthetics: Unified automatic quality assessment for speech, music, and sound. *arXiv preprint arXiv:2502.05139*, 2025.