UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

GERT WILLEM FOLZ

Analysis of Text-Conditioned Music Synthesis Models Generators

Work presented in partial fulfillment of the requirements for the degree of Bachelor in Computer Engineering

Advisor: Prof. Dr. Marcelo Soares Pimenta

Porto Alegre
March 2024

“Music is the universal language of mankind.”

— HENRY WADSWORTH LONGFELLOW

ACKNOWLEDGEMENTS

First and foremost, my deepest gratitude goes to my family. Their unwavering support and love have been the cornerstone of my journey. The values they instilled in me from a young age have shaped the person I have become and have been a constant source of strength and inspiration. I am forever grateful to them for their sacrifices, encouragement, and belief in my dreams.

I want to recognize and thank my advisors, Marcelo Soares Pimenta and Marcelo de Oliveira Johann, for their knowledge, guidance and mentorship throughout this work. Their insights and constructive feedback were fundamental to successfully conclude this research. My gratitude also goes out to the entire team from PCAD for their readiness to assist and support, granting me access to use their infrastructure and computational resources that were required for running the tests and experiments needed for this research. Without this assistance, I would not be able to complete this work properly.

Lastly, I would also like to extend my heartfelt thanks to all of my friends, inside and outside UFRGS, for all of their support and encouragement. I'd like to specially mention my dear friends Julio Costella Vicenzi, Bruno Bertoldi, and Jonas Bohrer. Their expertise and willingness to lend a hand during the technical challenges encountered in the experimental phase of this work have been invaluable. Their insights and assistance were not only instrumental in overcoming the hurdles but also enriched my understanding and approach to the problems faced. Without their help, this work would not have reached its fruition.

ABSTRACT

With the recent advancements in diffusion models, transformers, and the growing large-scale datasets, the field of generative models, particularly in the music-to-text context, has seen a remarkable surge in development and popularity. This thesis aims to conduct an extensive comparative analysis of the latest advancements in text-to-music models. The analysis will be structured around several key metrics to assess the effectiveness of each model, such as the quality of the generated audio and adherence to input text. Beyond these metrics, this analysis will delve into the underlying methodologies and technologies employed in each model, providing a comprehensive insight into the techniques and architectures driving the current state-of-the-art in text-to-music generation.

Keywords: Machine Learning in Music. Perceptual Audio Metrics. Text-to-Music Generation.

Análise de Modelos Generativos de Síntese Musical Condicionados por Texto

RESUMO

Com os recentes avanços na área de modelos de difusão, transformadores, e o crescimento de datasets de larga escala, o campo dos modelos generativos, particularmente na área de geração de texto-para-música, tem apresentado um aumento notável em desenvolvimento e popularidade. Esta tese tem como objetivo conduzir uma análise comparativa extensa dos últimos avanços dos modelos de texto-para-música. A análise será estruturada em torno de várias métricas chaves para avaliar a eficácia de cada modelo, como a qualidade do áudio gerado e a aderência ao texto de entrada. Além dessas métricas, esta análise se aprofundará nas metodologias e tecnologias subjacentes empregadas em cada modelo, fornecendo uma visão abrangente sobre as técnicas e arquiteturas que impulsionam o estado atual da arte na geração de música a partir de texto.

LIST OF ABBREVIATIONS AND ACRONYMS

TTM	Text-To-Music
AI	Artificial Intelligence
PLMs	Pre-Trained Language models
PCAD	Parque Computacional de Alto Desempenho
MLM	Masked Language Modelling
RVQ	Residual Vector Quantization
MIR	Music Information Retrieval
ML	Machine Learning
VAE	Variational Autoencoder
GAN	Generative Adversarial Network
SSL	Self Supervised Learning
CLAP	Contrastive Language-Audio Pretraining
FAD	Fréchet Audio Distance
KL	Kullback–Leibler Divergence
SDR	Signal to Distortion Ratio
SIR	Signal to Interference Ratio
FID	Fréchet Inception Distance

LIST OF FIGURES

Figure 2.1 The transformer architecture. 15
Figure 2.2 The diffusion process graphical model. 16
Figure 2.3 Left: A figure describing the VQ-VAE. Right: Visualisation of the embedding space. The output of the encoder $z(x)$ is mapped to the nearest point e_2. The gradient $\nabla_z L$ (in red) will push the encoder to change its output, which could alter the configuration in the next forward pass. 17
Figure 2.4 Generative adversarial nets are trained by simultaneously updating the **d**iscriminative distribution (D, blue, dashed line) so that it discriminates between samples from the data generating distribution (black, dotted line) p_x from those of the **g**enerative distribution p_g (G) (green, solid line). The lower horizontal line is the domain from which z is sampled, in this case uniformly. The horizontal line above is part of the domain of x. The upward arrows show how the mapping $x = G(z)$ imposes the non-uniform distribution p_g on transformed samples. G contracts in regions of high density and expands in regions of low density of p_g. (a) Consider an adversarial pair near convergence: p_g is similar to p_{data} and D is a partially accurate classifier. (b) In the inner loop of the algorithm D is trained to discriminate samples from data, converging to $D^*(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x)+p_g(x)}$. (c) After an update to G, gradient of D has guided $G(z)$ to flow to regions that are more likely to be classified as data. (d) After several steps of training, if G and D have enough capacity, they will reach a point at which both cannot improve because $p_g = p_{\text{data}}$. The discriminator is unable to differentiate between the two distributions, i.e. $D(x) = \frac{1}{2}$. 19

Figure 3.1 EnCodec: an encoder decoder codec architecture which is trained with reconstruction (ℓ_f and ℓ_t) as well as adversarial losses (ℓ_g for the generator and ℓ_d for the discriminator). The residual vector quantization commitment loss (ℓ_w) applies only to the encoder. 22
Figure 3.2 Illustration of the MERT Pre-training Framework. 23
Figure 3.3 Illustration of the CDPAM Pre-training Framework. 25

Figure 4.1 Learning framework diagram for MuLan. 28
Figure 4.2 Independent pre-training of the models providing the audio and text representations for MusicLM. 29
Figure 4.3 Codebook interleaving patterns for MusicGen. 30
Figure 4.4 ERNIE-Music overall architecture. 31
Figure 4.5 Illustration of the JEN-1 multi-task training strategy, including the text-guided music generation task, the music inpainting task, and the music continuation task. JEN-1 achieves the in-context learning task generalization by concatenating the noise and masked audio in a channel-wise manner. JEN-1 integrates both the bidirectional mode to gather comprehensive context and the unidirectional mode to capture sequential dependency. 32

Figure 5.1 The architecture of CLAP proposed model, including audio/text encoders, feature fusion, and keyword-to-caption augmentation. 36
Figure 5.2 FAD computation overview: using a pretrained audio classification model, VGGish, embeddings are extracted from both the output of a enhancement model that we wish to evaluate and a large database of background music. The Fréchet distance is then computed between multivariate Gaussians estimated on these embeddings. 37

LIST OF TABLES

Table 4.1 Models Information Summary 33
Table 4.2 Self-Reported Evaluation Metrics 33

Table 6.1 MusicGen generated music evaluation for FAD score and subjective music quality. Three different versions of the model are tested. 39
Table 6.2 Multiple embeddings used for the calculation of the FAD Score evaluation for the proposed models 41
Table 6.3 Comparison of the CLAP score and overall subjective music alignment with state-of-the-art text-to-music generation models 42

CONTENTS

1 INTRODUCTION **11**
2 BACKGROUND **14**
2.1 Transformers **14**
2.2 Diffusion Models **15**
2.3 Quantized Variational Autoencoders **16**
2.4 Generative Adversarial Networks **17**
2.5 Audio Representation **18**
2.5.0.1 Waveform 18
2.5.0.2 Spectrum 20
2.6 Single-task *vs.* Multi-task **20**
3 RELATED WORK **21**
3.1 EnCodec: High Fidelity Neural Audio Compression **21**
3.2 MERT: Acoustic Music Understanding Model with Large-Scale Self-supervised Training **22**
3.3 DPAM: A Differentiable Perceptual Audio Metric Learned from Just Noticeable Differences **23**
3.4 CDPAM: Contrastive Learning for Perceptual Audio Similarity **24**
3.5 DAC: High-Fidelity Audio Compression with Improved RVQGAN **25**
4 TEXT-TO-MUSIC MODELS **27**
4.1 MusicLM: Generating Music From Text **27**
4.2 Simple and Controllable Music Generation **28**
4.3 ERNIE-Music: Text-to-Waveform Music Generation with Diffusion Models **30**
4.4 JEN-1: TEXT-GUIDED UNIVERSAL MUSIC GENERATION WITH OMNIDIRECTIONAL DIFFUSION MODELS **31**
4.5 Models Summary **33**
5 STATE-OF-THE-ART METRICS **34**
5.1 CLAP **34**
5.2 Fréchet Audio Distance **34**
6 COMPARISON RESULTS **38**
6.1 FAD Score **38**
6.2 CLAP Score **41**
7 CONCLUSION **43**
7.1 Limitations **43**
7.2 Future Work **44**
REFERENCES **45**

1 INTRODUCTION

Recent years have witnessed remarkable progress in the field of deep generative artificial intelligence models, revolutionizing the way we create and interact with digital content. Central to this revolution are the generative models that can produce complex and diverse outputs from simple textual inputs. One prominent field that is calling the attention of the entire population is the text-to-image models, such as DALL-E (Ramesh et al., 2021), which have demonstrated an impressive ability to generate accurate visual representations from free textual descriptions. Inspired by the progress seen in text-to-image generation, researchers have been exploring and creating models (Kreuk et al., 2023; Yang et al., 2023) capable to generate audio from sequence-wide, high-level captions, such as "a windy breeze with birds singing in the background". A common technique used in these models is casting audio synthesis as a language modeling task in a discrete representation space, and leveraging a hierarchy of coarse-to-fine audio discrete tokens (Liu et al., 2023). These models have not only opened new possibilities in digital art creation but also have significant implications in fields like graphic design and visual communication, besides the extended applications that generating audio can have, such as sound effects for the creation of media entertainment, showing the potential of generative models in multiple fields.

Generating high-fidelity music from text descriptions, like "An 80s synthpop song with slow pace and heavy drums in the background", is a challenging task that requires modeling long range sequences called text-to-music (TTM) generation. Firstly, unlike speech recordings that use lower sampling rates (e.g. 16kHz), music requires sampling the signal at a higher rate, like 44.1KHz or 48 kHz, to capture the necessary intricacies. Secondly, music can contain multiple instruments and an arrangement of melodies, creating a very complex structure. Considering that human listeners are extremely sensitive to disharmony and dissonance, melodic errors introduced in the process of music generation are extremely noticeable. Finally, having the possibility to control attributes that music impainting and music continuation offers, like key, timbre, and melody, is crucial for music creators.

Recent advancements in self-supervised audio representation learning (Balestriero et al., 2023), sequential modeling (Touvron et al., 2023), audio synthesis (Tan et al., 2021) and music information retrieval (MIR) have greatly influenced the way we interact and create sound and music, providing the conditions to develop models (Agostinelli et al.,

2023; Copet et al., 2023; Li et al., 2023a; Zhu et al., 2023) capable of synthesizing music given a free-form textual input, leading to significant advancements in the fields of audio processing and music generation. This thesis explores these models, underpinned by the influence of machine learning (ML) and artificial intelligence (AI) techniques.

In this research, we delve into the nuanced complexities of generating music using AI, the innovative algorithms and architectures used that are thriving in music generation, besides exploring the weaknesses and robustness of the evaluation metrics that align closely with human auditory perception commonly employed for the comparison between the state-of-the-art models. However, those models still face multiple obstacles to generate satisfactory results. Besides the inherent difficulty of synthesizing high-quality and coherent music, this difficulty is further increased by the scarcity of paired music-text data, a crucial resource for training generative models. This situation stands in contrast to the field of text-to-image generation, where the availability of extensive datasets has contributed immensely to the advancements achieved in recent years. One of the reasons for this is that creating text descriptions of general music is not as straightforward as describing images. Besides not being easy to unambiguously capture with just a few words the salient characteristics of music (e.g., the melody, the rhythm, the timbre of vocals and the many instruments used in accompaniment), the audio is also structured along a temporal dimension which makes sequence-wide captions a much weaker level of annotation than an image caption.

Our comprehensive experiments reveal a critical insight through the robustness of objective metrics commonly employed in evaluating TTM generative process. These metrics, we find, often fail to align accurately with human auditory perception. This discrepancy highlights a significant gap in the current evaluation methodologies for these models. We conduct an extensive evaluation of the audio quality and the coherence between the input text and generated music created by these models, emphasizing the critical need of considering how the different baselines used in the comparison for music quality affects the evaluation between models, and also how various embeddings employed to extract the features and characteristics from audio samples can impact on the overall analysis of a specific model.

In summary, the key contributions of this work are:

1. We extensively evaluate the audio quality generated by the state-of-the-art text-to-music models.
2. We demonstrate how the popular metrics currently in use fail to predict the percep-

tual quality of generative music.

3. We propose a more accurate and reliable approach on how to evaluate the quality of the output generated by these models that properly aligns with the human auditory perception.

2 BACKGROUND

In this section, we provide an overview of the existing literature in the field of music generation, reviewing some important concepts that are necessary for a better understanding for this work and also highlighting some technologies and techniques involved in the development of TTM models.

2.1 Transformers

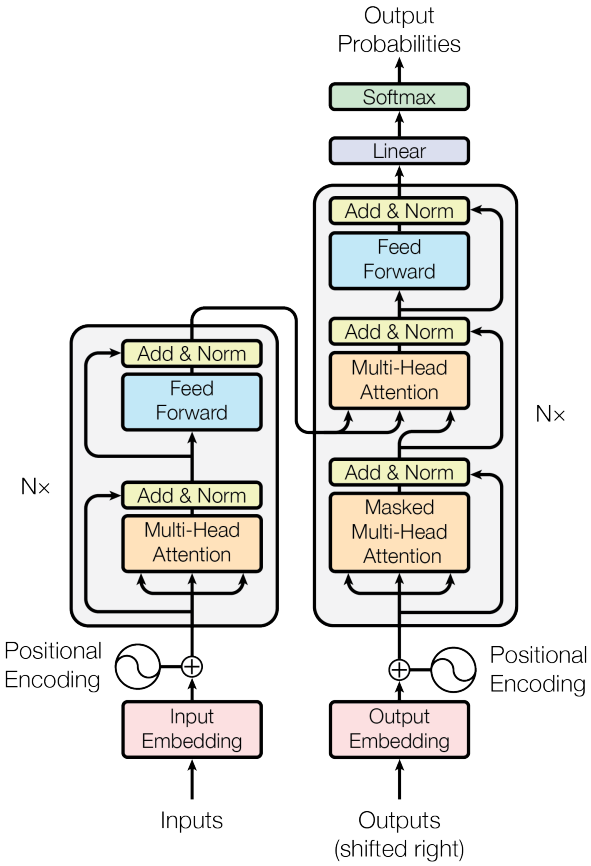
A transformer (Vaswani et al., 2017) is a type of neural network architecture predominantly used in the field of natural language processing (NLP). Unlike traditional neural network architectures that process data sequentially, transformers utilize a mechanism called attention, or self-attention, that enables the model to consider the entire input sequence at once, allowing it to capture complex relationships between different parts of the sequence, which is particularly useful in understanding the context and semantics in language tasks. In essence, they allow the model to focus on different parts of the input sequence when making predictions.

Transformers consist of two main parts: the encoder and the decoder. The encoder processes the input data (like a sentence in a language translation task) and generates a context-rich representation of it, called an embedding. The decoder then uses this representation to generate the output data (like the translated sentence). This architecture makes transformers highly effective for a range of tasks such as machine translation, text summarization, question-answering, etc. The overall architecture of a transformer is exemplified in figure 2.1.

The transformer model's ability to process inputs in parallel significantly improves training efficiency over prior models that required sequential data processing. This parallelization has been key to the model's scalability and effectiveness in handling large datasets and complex tasks. In the context of music generation, transformers generate music by predicting the next note based on the previous ones. Models like MusicLM(Agostinelli et al., 2023), MusicGen (Copet et al., 2023) and JEN-1 (Li et al., 2023a) employ transformer-based decoder-only models to autoregressively generate audio tokens in the music sequence. Such autoregressive models can produce highly coherent audio as each token generation is conditioned on the previous context. However, the sequential token-by-token generation manner inherently sacrifices speed for both generation and in-

ference, restricting the applicability of such techniques in downstream tasks.

Figure 2.1 – The transformer architecture.



Source: (Vaswani et al., 2023)

2.2 Diffusion Models

A diffusion model (Sohl-Dickstein et al., 2015) is class of generative AI models that works by gradually adding Gaussian noise to the original data in the forward diffusion process and then learning to remove the noise in the reverse diffusion process. It is a latent variable model referring to a hidden continuous feature space, it behaves similarly to VAEs (Variational Autoencoders) and is inspired on non-equilibrium thermodynamics.

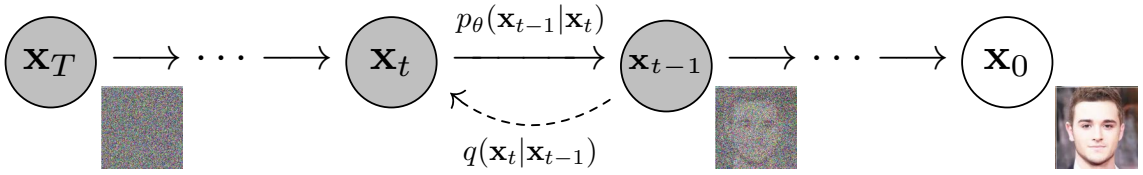
In the forward diffusion process, the model gradually adds Gaussian noise to the input $\mathbf{x}$ through a series of T steps. Firstly, starts with sampling a data point $\mathbf{x}_0$ from the real data distribution $q(\mathbf{x})$ like $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ and then adding some Gaussian noise with variance β_t to $\mathbf{x}_{t-1}$, producing a new latent variable $\mathbf{x}_t$ with distribution $q(\mathbf{x}_t|\mathbf{x}_{t-1})$. The forward diffusion process is represented in equation 2.1.

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0)\prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}), \qquad q(\mathbf{x}_t|\mathbf{x}_{t-1})\mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \tag{2.1}$$

The reverse diffusion process is the process of training a neural network to recover the original data by reversing the noising process applied in the forward pass, as shown in equation 2.2. Estimating $q(\mathbf{x}_t|\mathbf{x}_{t-1})$ is difficult as it can require the whole dataset. That's why a parameterized model p_θ can be used to learn the parameters. For small enough β_t it will be a Gaussian and can be obtained by just parameterizing the mean and variance. This entire process of forward and reverse diffusion is illustrated in figure 2.2.

$$p_\theta(\mathbf{x}_{0:T})p(\mathbf{x}_T)\prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t), \qquad p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)\mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t)) \tag{2.2}$$

Figure 2.2 – The diffusion process graphical model.



Source: (Ho; Jain; Abbeel, 2020)

In the context of music generation, diffusion models can generate music in parallel, offering faster generation speed but often at the cost of lower coherence. The trade-off between generation speed and coherence remains a key challenge in this area.

2.3 Quantized Variational Autoencoders

Quantized Variational Autoencoders (VQ-VAE) represent an advanced class of generative models that integrate the principles of variational autoencoders (VAEs) with vector quantization techniques (Oord; Vinyals; Kavukcuoglu, 2018). VQ-VAE models are particularly effective in handling tasks where discrete representations are more suitable, such as in music, speech and image generation, where we first need to convert the input into discrete tokens.

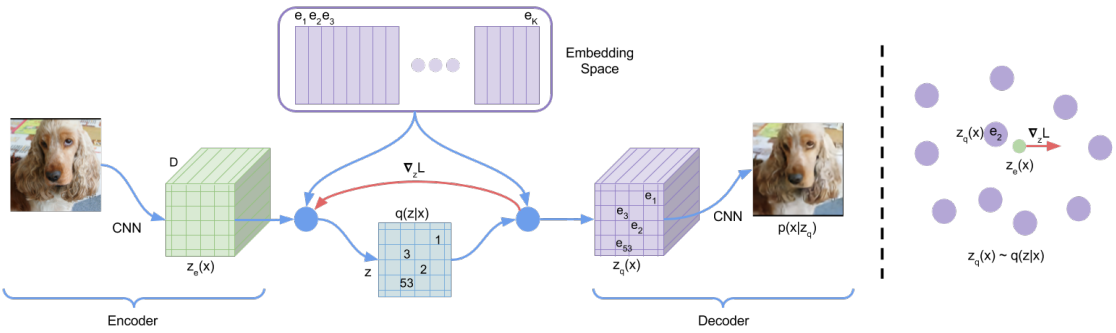
In a VQ-VAE, the encoder first maps the input data $\mathbf{x}$ to a continuous latent space. The key aspect of VQ-VAE is the quantization step, where the continuous latent represen-

tation is converted into a discrete form. This is achieved by mapping each vector in the latent space to the nearest vector in a predefined set of vectors, known as a codebook. The codebook contains a finite set of vectors, and each vector is referred to as a codeword. The process of mapping to the nearest codeword can be represented by equation 2.3.

$$z_q(x) = e_k, \quad \text{where} \quad k = \text{argmin}_j \|z_e(x) - e_j\|_2 \tag{2.3}$$

The decoder in VQ-VAE then takes these quantized latent vectors and reconstructs the input data. The quantization step introduces a discrete bottleneck in the model, which forces the latent space to learn a more structured and efficient representation of the data. The reconstruction loss, combined with a commitment loss that keeps the encoder outputs close to the chosen codeword, is used to train the model. This architecture allows VQ-VAE to generate high-quality and diverse samples while maintaining computational efficiency. The process of encoding, quantization, and decoding in VQ-VAE is illustrated in figure 2.3.

Figure 2.3 – Left: A figure describing the VQ-VAE. Right: Visualisation of the embedding space. The output of the encoder $z(x)$ is mapped to the nearest point e_2. The gradient $\nabla_z L$ (in red) will push the encoder to change its output, which could alter the configuration in the next forward pass.



Source: (Oord; Vinyals; Kavukcuoglu, 2018)

2.4 Generative Adversarial Networks

Generative Adversarial Networks (GANs) are a class of generative models that have gained significant attention for their ability to generate highly realistic data (Goodfellow et al., 2014). The generative model in a GAN is pitted against an adversary: a discriminative model that learns to determine whether a sample is from the model distribution or the data distribution. The generative model can be thought of as analogous

to a team of counterfeiters, trying to produce fake currency and use it without detection, while the discriminative model is analogous to the police, trying to detect the counterfeit currency. Competition in this game drives both teams to improve their methods until the counterfeits are indistinguishable from the genuine articles.

In summary, the generator network in a GAN learns to generate data by mapping latent space vectors, drawn from a prior distribution, to the data space. The generated data is intended to mimic the real data distribution. The discriminator, on the other hand, is trained to distinguish between the real data and the data generated by the generator. This training process can be conceptualized as a minimax game, where the generator tries to fool the discriminator, and the discriminator tries to correctly classify real and fake data, as represented in equation 2.4.

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]. \tag{2.4}$$

The training of GANs involves back-and-forth optimization of the generator and the discriminator. The generator learns to produce more realistic data, while the discriminator becomes better at identifying the generated data. This adversarial process continues until the generator produces data indistinguishable from real data, at which point the discriminator is maximally confused. GANs have been successfully applied in various domains, including image synthesis, style transfer, and super-resolution. The adversarial training process and the interaction between the generator and discriminator in GANs are depicted in figure 2.4.

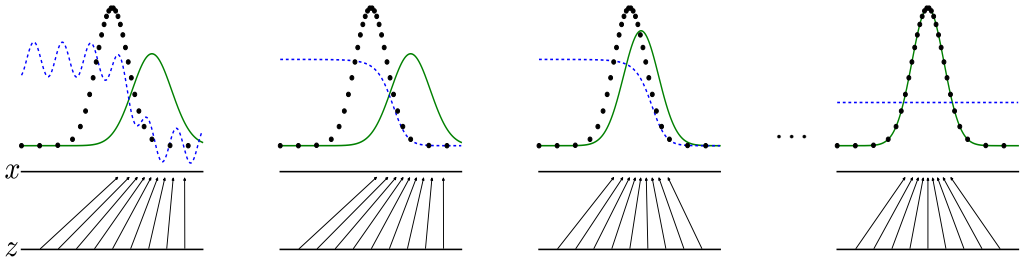
2.5 Audio Representation

Audio representation in digital signal processing and machine learning can be primarily categorized into two forms: waveform and spectrum. These representations are fundamental in various audio-related applications, including speech recognition, music information retrieval, music generation and audio synthesis.

2.5.0.1 Waveform

The waveform representation of audio is a direct depiction of sound waves as they vary over time. In digital audio, a waveform is typically represented as a series of discrete

Figure 2.4 – Generative adversarial nets are trained by simultaneously updating the **d**iscriminative distribution (D, blue, dashed line) so that it discriminates between samples from the data generating distribution (black, dotted line) p_x from those of the **g**enerative distribution p_g (G) (green, solid line). The lower horizontal line is the domain from which z is sampled, in this case uniformly. The horizontal line above is part of the domain of x. The upward arrows show how the mapping $x = G(z)$ imposes the non-uniform distribution p_g on transformed samples. G contracts in regions of high density and expands in regions of low density of p_g. (a) Consider an adversarial pair near convergence: p_g is similar to p_{data} and D is a partially accurate classifier. (b) In the inner loop of the algorithm D is trained to discriminate samples from data, converging to $D^*(x) = \frac{p_{\text{data}}(x)}{p_{\text{data}}(x)+p_g(x)}$. (c) After an update to G, gradient of D has guided $G(z)$ to flow to regions that are more likely to be classified as data. (d) After several steps of training, if G and D have enough capacity, they will reach a point at which both cannot improve because $p_g = p_{\text{data}}$. The discriminator is unable to differentiate between the two distributions, i.e. $D(x) = \frac{1}{2}$.



(a) (b) (c) (d)

Source: (Goodfellow et al., 2014)

amplitude values sampled at regular intervals. This form of representation captures the temporal dynamics of sound and is particularly useful for tasks that require a detailed understanding of the temporal structure of audio, such as time-domain processing and audio editing.

Waveform representation maintains the raw, untransformed state of sound, making it suitable for tasks where preserving the original audio fidelity is crucial. Considering the computational efficiency, using raw audio waveforms as model inputs or generation targets is extremely challenging, owing to the high complexity of waveform signals.

This approach uses quantization-based audio codecs, like SoundStream (Zeghidour et al., 2021) or EnCodec (Défossez et al., 2022a), to tokenize the continuous waveform into a compact, compressed, and discrete representation, while maintaining high reconstruction quality. For example, MusicGen (Copet et al., 2023) puts a transformer decoder over EnCodec quantized units, conditioned on text or melody. AudioLM (Borsos et al., 2023) and AudioPaLM (Rubenstein et al., 2023) take text, decode it into audio tokens via a transformer, then convert the tokens back to raw audio using SoundStream.

2.5.0.2 Spectrum

Spectral representation, on the other hand, transforms the audio signal from the time domain to the frequency domain, often using techniques like the Fast Fourier Transform. This transformation results in a spectrum that shows how the energy of the audio signal is distributed across different frequency components.

Spectral representations, such as mel-spectrograms, are particularly useful in applications that require analysis of the frequency content of audio signals, such as in speech recognition, music genre classification, and environmental sound analysis. By representing audio in the frequency domain, spectral analysis can reveal insights about the harmonic structure, timbre, and other characteristics that are not readily apparent in the waveform representation.

This approach first converts the waveform into a mel-spectrogram and then processes it by referencing techniques from computer vision, using vector quantized variational autoencoders (VQ-VAE) or generative adversarial networks (GANs). For example, Diffwave (Kong et al., 2020) and Diffsound (Yang et al., 2023) feed textual tags or other conditional signals into a spectrogram decoder to generate mel-spectrogram tokens. The tokens are fed into a pre-trained audio VQ-VAE to synthesize the mel-spectrogram, which is finally converted into the audio waveform through a vocoder like HiFi-GAN (Kong; Kim; Bae, 2020).

2.6 Single-task *vs.* Multi-task

Conditional neural music generation can be categorized into two types, Multi-task and Single-task. The former uses low-level control signals with tight temporal alignment to the audio output, which includes lyrics-conditioned music generation and audio synthesis from MIDI sequences, with tight temporal alignment to the audio output. The latter utilizes high-level semantic descriptions, like text (Kreuk et al., 2023), as conditioning signals, where it provides overall coherence and consistency without precise temporal alignment. However, in practical applications, pairs such as <conditional signal, audio> are often scarce. Hence, models are commonly trained on unlabeled audio datasets using self-supervised techniques to boost generalization.

3 RELATED WORK

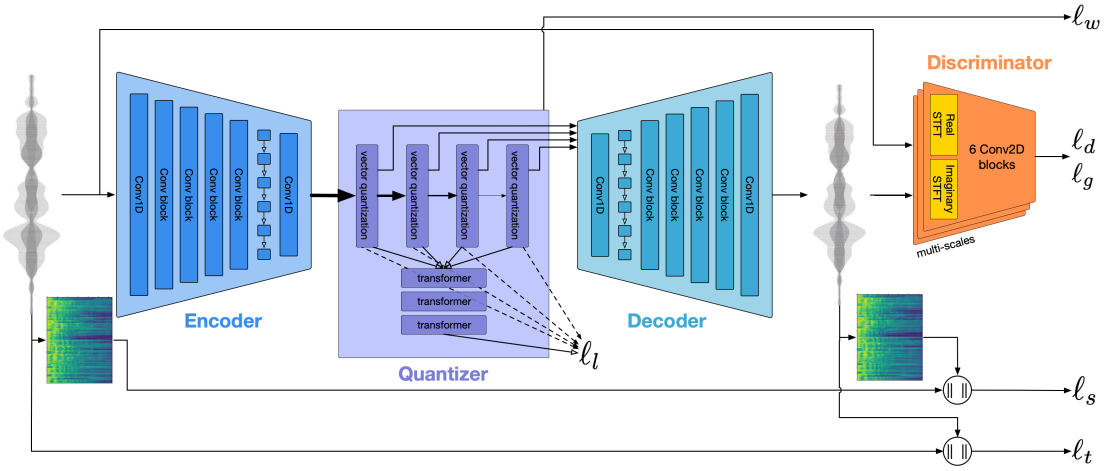
In this section, we provide an overview of the existing literature in the field of audio synthesis and other topics related to music generation, reviewing some important papers that model music information into quantized tokens, explore how these models generate different embbedings and their corresponding architectures, as well as the implementation techniques used by them.

3.1 EnCodec: High Fidelity Neural Audio Compression

EnCodec (Défossez et al., 2022a) is a streaming convolutional based encoder-decoder architecture with a sequential modeling component. This architecture is applied to both the encoder and decoder sides of the model and is composed of three main components: An encoder network E is input an audio extract and outputs a latent representation z; The second part is a quantization layer Q that produces a compressed representation z_q, using Residual Vector Quantization (RVQ). This process involves projecting input vectors onto the nearest entries in a codebook and then further refining this quantization by using additional codebooks for the residual quantization. This quantization process converts the continuous latent representation into a discrete set of indices, which can be re-transformed into a vector form before being fed into the decoder. Finally, a decoder network G reconstructs the time-domain signal, $\hat{x}$, from the compressed latent representation z_q. The whole system is trained end-to-end to minimize a reconstruction loss applied over both time and frequency domain, together with a perceptual loss in the form of discriminators operating at different resolutions. A visual description of this method is illustrated in Figure 3.1.

The EnCodec model simplifies and accelerates training through a multi-scale spectrogram adversary, effectively reducing artifacts and generating superior audio samples. A new loss balancer mechanism stabilizes training by adjusting the weight of each loss term to define its contribution to the overall gradient. This architecture also incorporates lightweight Transformer models for further compression of the audio representation, enhancing efficiency while maintaining real-time processing capabilities. The codec has been extensively evaluated across various audio domains, including speech, noisy-reverberant speech, and music, demonstrating superior performance over traditional methods for both 24 kHz monophonic and 48 kHz stereophonic audio.

Figure 3.1 – EnCodec: an encoder decoder codec architecture which is trained with reconstruction (ℓ_f and ℓ_t) as well as adversarial losses (ℓ_g for the generator and ℓ_d for the discriminator). The residual vector quantization commitment loss (ℓ_w) applies only to the encoder.



Source: (Défossez et al., 2022b)

3.2 MERT: Acoustic Music Understanding Model with Large-Scale Self-supervised Training

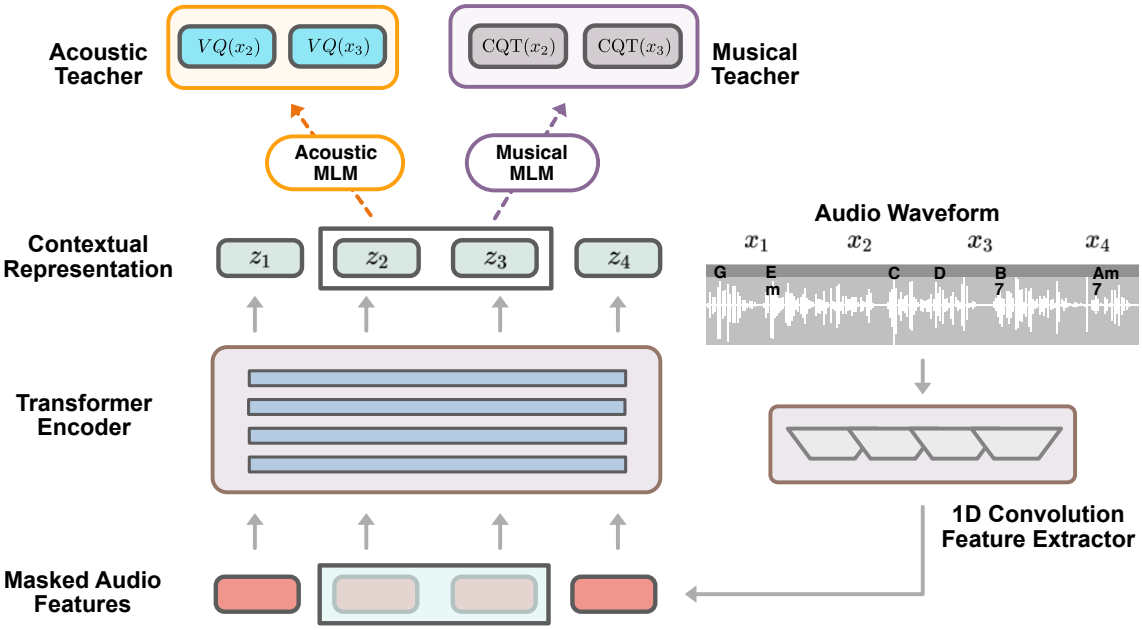
MERT (Li et al., 2023b) is an open-source pre-trained acoustic music model that is built on the foundation laid by pre-trained language models (PLMs), which have shown remarkable success in learning generalizable representations of data in various fields, including natural language processing. Recognizing music as a special language, MERT adapts PLM-based methods to model music sequences. This approach aims to unify the modeling of a wide range of music understanding tasks, also known as Music Information Retrieval (MIR), including music tagging, beat tracking, music transcription, source separation, etc., leveraging the inherent similarities between music and language as communication interfaces so that different tasks no longer need detailed models or features, and also intending to use a PLM for acoustic music understanding can re-distribute the musical knowledge rather than the data itself.

MERT's methodology involves a pre-training paradigm that includes prediction to acoustic teachers and reconstruction to music teachers, both anchored in the established masked language model (MLM) paradigm. This structure allows for a nuanced approach to modeling both acoustic and musical information in audio data. For acoustic information modeling, MERT uses two approaches: traditional features and deep learning-based features. The traditional method employs k-means clustering on log-Mel spectrum and Chroma features for timbre and harmonic acoustic information, and the deep learning

approach utilizes EnCodec (Défossez et al., 2022a). For the music teachers, MERT incorporates a reconstruction loss to the Constant-Q Transform (CQT) (Brown, 1991) spectrogram, emphasizing pitch-level information crucial for tasks like pitch detection, chord recognition, and music transcription. This approach utilizes mean squared error (MSE) loss for reconstruction, offering a more nuanced understanding of the musical aspects of audio.

MERT inherits a speech self supervised learning (SSL) paradigm, employing teacher models to generate pseudo targets for sequential audio clips. MERT incorporates a multi-task paradigm to balance the acoustic and musical representation learning to capture the distinctive pitched and tonal characteristics in music, as shown in Fig. 3.2.

Figure 3.2 – Illustration of the MERT Pre-training Framework.



Source: (Li et al., 2023b)

3.3 DPAM: A Differentiable Perceptual Audio Metric Learned from Just Noticeable Differences

DPAM (Manocha et al., 2020) is based on just noticeable differences (JNDs) – the minimal change at which a difference is perceived, and intends to bridge the gap between human judgment and automated audio evaluation metrics. The model is trained on a large-scale dataset of human judgments wherein subjects are asked whether two audio recordings sound the same or different. Recordings are modified by injecting various perturbations characteristic of degradations commonly found in audio processing tasks,

like noise, reverb, and compression artifacts, and the goal is to determine the JND threshold for each subject. The data collection is optimized using active learning strategies, efficiently gathering labeled data that are later used to train a perceptual metric.

The architecture is based on (Germain; Chen; Koltun, 2018) consisting of 14 convolutional layers with 3×1 kernels, batch normalisation and leaky ReLU units, and zero padding to reduce the output dimensions by half after every step. Furthermore, the authors also publicly release the dataset, code and resulting metric, as well as listening test examples.

3.4 CDPAM: Contrastive Learning for Perceptual Audio Similarity

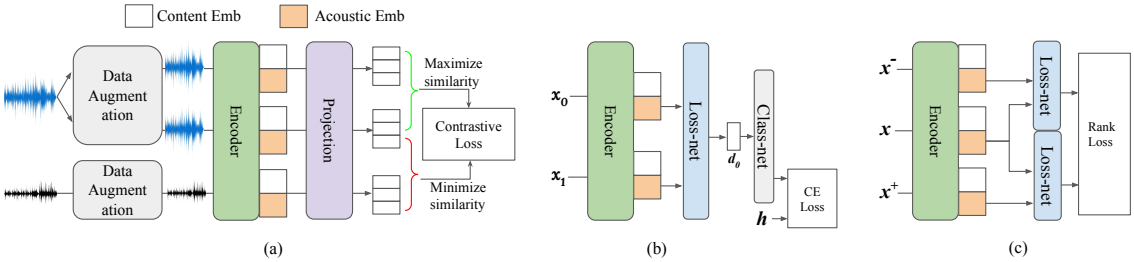
CDPAM (Manocha et al., 2021) is a new approach in perceptual audio metric, building upon the DPAM (Manocha et al., 2020) model. The DPAM model suffers from a natural tension between the cost of data acquisition and generalization beyond that data. It requires a large set of human judgments to span the space of perturbations in which it can robustly compare audio clips. Moreover, the metric may generalize poorly to unseen content. Lastly, because the data is focused near JNDs, it is likely to be less robust to large audio differences.

To circumvent these limitations, besides also focusing on just noticeable differences (JNDs), CDPAM uses contrastive learning, multi-dimensional representation learning, and triplet learning to improve the robustness and generalizability of the metric across a range of audio perturbations. Multidimensional representation learning is used to separately model content similarity and acoustic similarity. The combination of contrastive learning and multi-dimensional representation learning allows CDPAM to better generalize across content differences with limited human annotation. To further improve robustness to large perturbations beyond JND, the authors collect a dataset of judgments based on triplet comparisons, asking subjects: “Is A or B closer to reference C?”

The architecture of CDPAM consists of an audio encoder, a projection network, and a loss network. The audio encoder outputs two sets of embeddings: acoustic and content, which are then processed through the projection network. The loss network, trained on JND data and fine-tuned with triplet comparison data, outputs a distance that predicts human judgment. This design allows CDPAM to effectively differentiate between various audio perturbations and to generalize well to unseen audio content. The architecture of CDPAM model is illustrated in figure 3.3.

CDPAM has been validated across nine diverse datasets, showing better correlation with MOS and triplet comparison tests than its predecessor, DPAM. Its application to tasks like speech synthesis and enhancement has demonstrated significant improvements, highlighting the model's potential in various audio processing applications. This model presents a significant advancement in the field of audio quality assessment, aligning more closely with human auditory perception than traditional metrics.

Figure 3.3 – Illustration of the CDPAM Pre-training Framework.



Source: (Manocha et al., 2021)

3.5 DAC: High-Fidelity Audio Compression with Improved RVQGAN

DAC (Kumar et al., 2023) is an audio compression model designed to compress 44.1 KHz audio into discrete codes at an 8kbps bitrate, achieving approximately 90x compression with minimal quality loss and reduced artifacts. It addresses key challenges in the field of audio compression, including codebook collapse and quantizer dropout, and introduces significant innovations like periodic activation functions, enhanced residual vector quantization, and multi-scale STFT discriminators. This model demonstrates its versatility by handling diverse audio types such as speech, music, and environmental sounds at different sampling rates and formats, making it a universal solution for high-fidelity audio compression.

The model is built on the framework of VQ-GANs and uses the fully convolutional encoder-decoder network from SoundStream (Zeghidour et al., 2021), that performs temporal downscaling and quantize the encodings using Residual Vector Quantization (RVQ) with factorized and L2-normalized codes to overcome the limitations of traditional vector quantization, significantly improving codebook usage and bitrate efficiency. The former decouples code lookup and code embedding, and the latter converts euclidean distance to cosine similarity, which is helpful for stability and quality, both used to significantly improve codebook usage, increasing bitrate efficiency and reconstruction quality.

In addition to that, the model presents novel techniques and key features, such as:

Periodic Activation Function. The model employs the Snake activation function (Ziyin; Hartwig; Ueda, 2020) to add periodic inductive bias, improving the handling of periodic signals in audio waveforms, thus enhancing fidelity.

Quantizer Dropout Rate. Quantizer dropout was introduced in SoundStream (Zeghidour et al., 2021) to train a single compression model with variable bitrate, however, applying quantizer dropout degrades the audio reconstruction quality at full bandwidth. A modified approach to quantizer dropout is adopted, setting the dropout probability at 0.5 to balance the reconstruction quality across various bitrates.

Discriminator Design. Using magnitude spectrograms, like prior works, discards phase information which can be utilized by the discriminator to penalize phase modeling errors. Moreover, high-frequency modeling is still challenging for these models especially at high sampling rates. Utilizing a complex STFT discriminator (Zeghidour et al., 2021) at multiple time-scales, the model effectively addresses high-frequency modeling challenges and reduces aliasing artifacts, besides improving phase modeling.

Loss Functions. The model combines frequency domain reconstruction loss and adversarial loss, employing a multi-scale approach to mel-spectrogram loss calculation and using straightforward codebook and commitment losses without complex loss balancing.

4 TEXT-TO-MUSIC MODELS

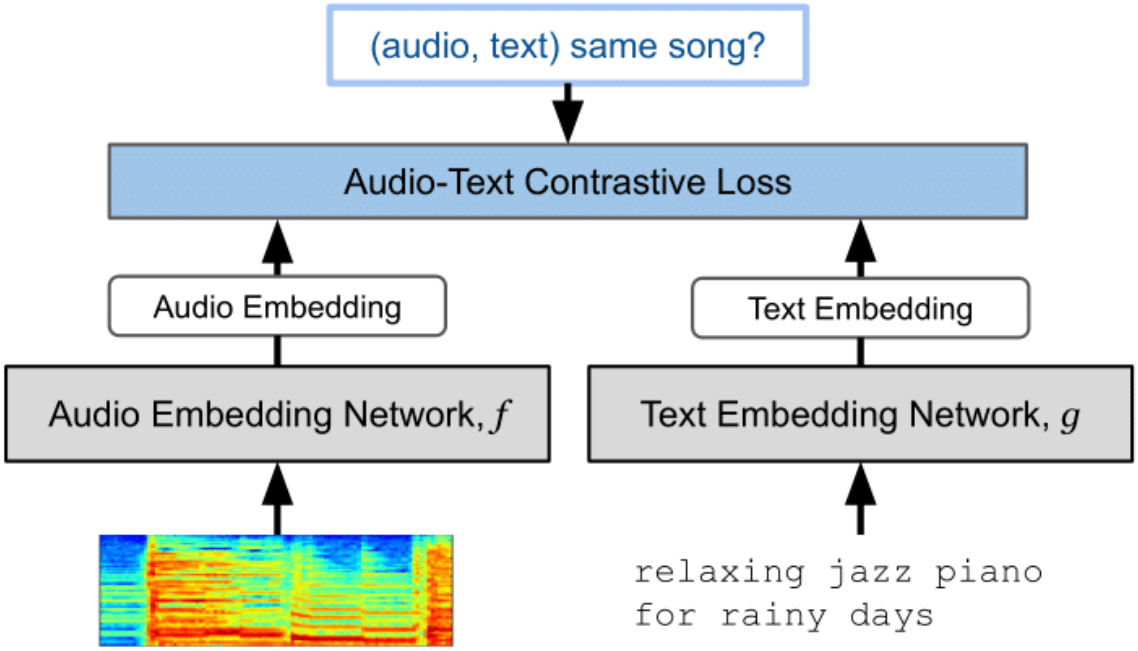
This chapter delves into the details of Text-to-Music (TTM) models, an intersection of language processing and music generation. Leveraging recent advancements in machine learning and artificial intelligence, TTM models are capable of transforming textual descriptions into rich, nuanced musical compositions. These models represent a significant leap in both natural language understanding and creative AI, enabling a new form of expression where text and music coalesce. We explore the most prominent models in this field, their unique approaches and technologies to this emerging area. From employing hierarchical sequence-to-sequence modeling to integrating diffusion models, these TTM systems display a novel approach in music creation and audio processing. We will explore the models MusicLM (Agostinelli et al., 2023), MusicGen (Copet et al., 2023), ERNIE-Music (Zhu et al., 2023), and JEN-1 (Li et al., 2023a), showcasing distinct methodologies and capabilities in generating music from text.

4.1 MusicLM: Generating Music From Text

MusicLM (Agostinelli et al., 2023) proposes casting the process of conditional music generation as a hierarchical sequence-to-sequence modeling task. To achieve this, MusicLM uses AudioLM's (Liu et al., 2023) multi-stage autoregressive modeling as the generative component, while extending it to incorporate text conditioning. To address the challenge of text-music pairs data scarcity, MusicLM incorporates MuLan (Huang et al., 2022). MuLan is a music-text joint embedding model consisting of two embedding towers, one for each modality. This allows the authors to easily scale the training data and to increase the robustness of noisy text descriptions. MuLan architecture is showed on figure 4.1.

MusicLM architecture is composed of three independently pre-trained models for extracting audio representations that will serve for conditional autoregressive music generation: SoundStream (Zeghidour et al., 2021), an audio compressor which receives an waveform as input and compresses it at a lower bit rate, maintaining a high reconstruction quality used for generating acoustic tokens, making use of the residual vector quantization (RVQ) technique; the masked-language-modeling (MLM) module of a w2v-BERT (Chung et al., 2021) for semantic tokens to maintain long-term coherent generation; MuLan embeddings computed from the audio as conditioning during training, and MuLan

Figure 4.1 – Learning framework diagram for MuLan.



Source: (Huang et al., 2022)

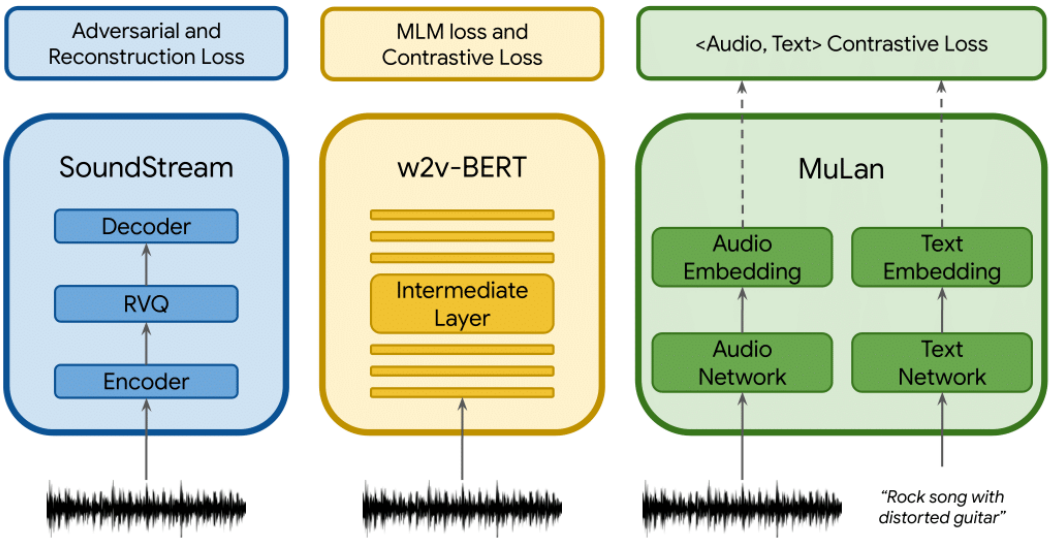
embeddings computed from the text input during inference. These components are illustrated in Figure 4.2.

To support future research, the authors also publicly release MusicCaps, a high-quality music caption dataset hand-curated with 5.5k text-music pairs prepared by expert musicians. The authors also extend the generative process to include other conditioning signals beyond text, extending MusicLM to accept an additional melody in the form of audio (e.g., whistling, humming) as conditioning to generate music that follows the desired melody, rendered in the style described by the text prompt.

4.2 Simple and Controllable Music Generation

MusicGen (Copet et al., 2023) is a controllable music generation model able to generate music given textual description, consisting of a single language model comprised of a single-stage transformer that operates over several streams of compressed discrete music representation, eliminating the need for cascading several models, such as hierarchically and upsampling. The language model is over the quantized units from EnCodec (Défossez et al., 2022b), a convolutional auto-encoder with a latent space quantized using Residual Vector Quantization (RVQ) (Zeghidour et al., 2021), and an adversarial reconstruction loss. Compression models that employ Residual Vector Quantization (RVQ)

Figure 4.2 – Independent pre-training of the models providing the audio and text representations for MusicLM.



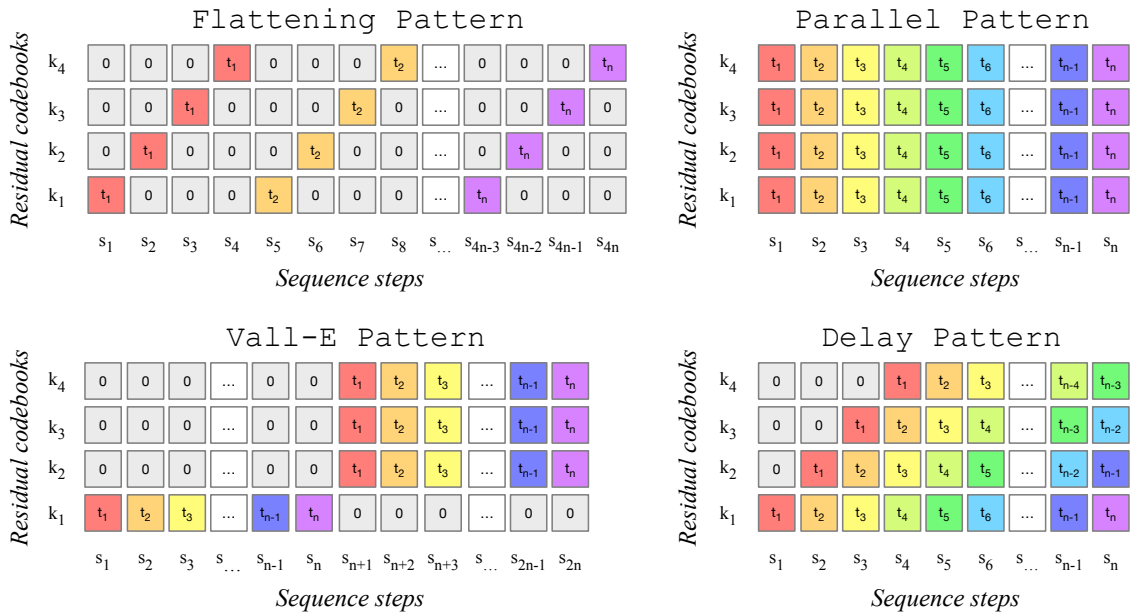
Source: (Agostinelli et al., 2023)

results in several parallel streams, where each stream is comprised of discrete tokens originating from different learned codebooks. A codebook refers to the set of quantized values that represent the audio signal generated by the audio tokenization model EnCodec.

The authors introduce a new modeling framework, which generalizes to various codebook interleaving patterns, and explore several variants. Through patterns, the authors can leverage the internal structure of the quantized audio tokens. In RVQ, each quantizer encodes the quantization error left by the previous quantizer, thus quantized values for different codebooks are in general not independent, and the first codebook is the most important one. Through empirical evaluations, the authors showed the benefits and drawbacks of various codebook patterns, such as exact flattened autoregressive decomposition and inexact autoregressive decomposition, as it's illustrated on Figure 4.3.

The authors also show that generally there are three main approaches for representing text for conditional audio generation. (Kreuk et al., 2023) proposed using a pretrained text encoder, specifically T5 (Raffel et al., 2020). (Chung et al., 2022) show that using instruct-based language models provide superior performance and finally, (Agostinelli et al., 2023; Liu et al., 2023; Huang et al., 2023; Sheffer; Adi, 2023) claimed that joint text-audio representation, such as CLAP (Wu et al., 2023), provides better-quality generations. The authors experiment with all of the above. Besides text conditioning music generation, MusicGen also allows the conditioning on a melodic structure from another audio track, whistling or humming. Such an approach also allows for an

Figure 4.3 – Codebook interleaving patterns for MusicGen.



Source: (Copet et al., 2023)

iterative refinement of the model's output.

4.3 ERNIE-Music: Text-to-Waveform Music Generation with Diffusion Models

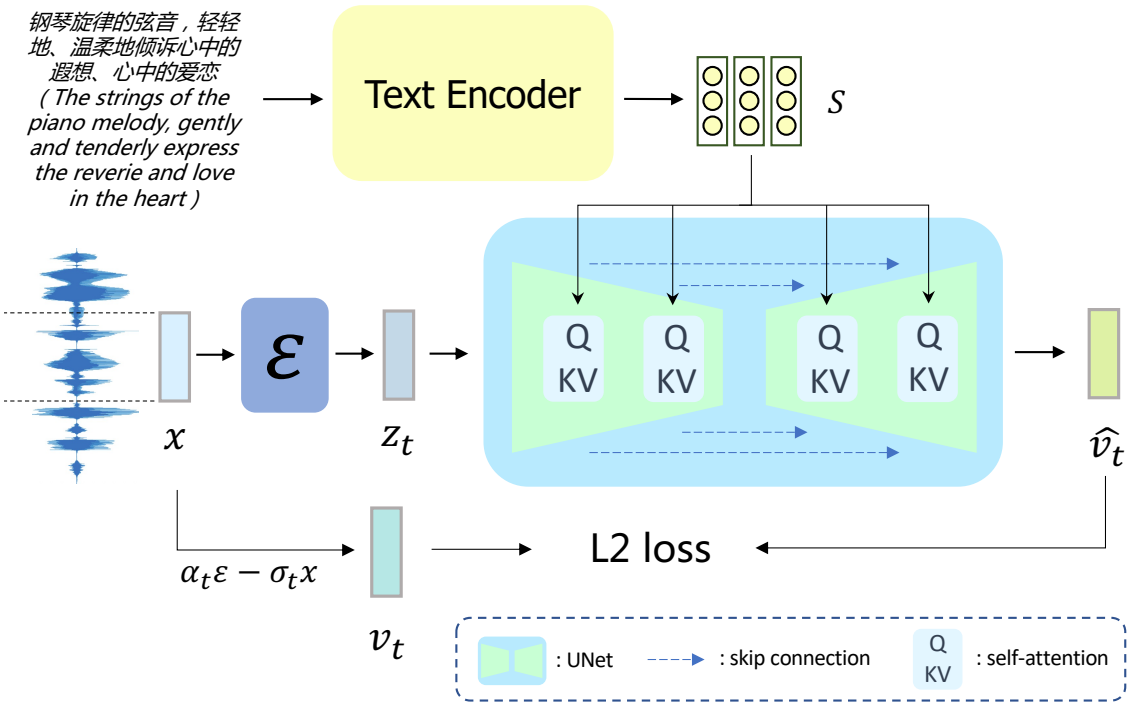
ERNIE-Music (Zhu et al., 2023) was the first text-to-music generator model that can receive as an input a free-form textual prompt as the condition to guide the waveform generation process using diffusion models. To circumvent the problem of the scarcity of a large text-to-music dataset, (Zhu et al., 2023) collects the data from the Internet on music service supporting platforms by utilizing the "comment voting" mechanism, where users rate each other comments via the "upvote" option. The authors consider the "popular comments" as generally relatively high quality and usually contain much useful music-related information such as musical instruments, genres, and expressed human moods.

As shown in Figure 4.4, ERNIE-Music overall model architecture contains a conditional music diffusion model which models the predicted *velocity* $\hat{v}_\theta(z_t, t, y)$, and a text encoder $\mathrm{E}(\cdot)$ that maps text tokens into a sequence of vector representations. The inputs of the music diffusion model are latent variable z_t, timestep t, and the representation of text sequence. The output is the estimated *velocity* $\hat{v}_t$. The authors adopt the architecture of UNet whose key components are stacked convolutional blocks and self-attention blocks, which model the global information of the music signals. Generation models can estimate the conditional distribution and the conditional information can be fused into the

generative models in many ways.

The authors also study which format of text used as input benefits the model to learn text-music relevance, comparing between generating music based on a set of pre-defined music tags representing the specific music's feature, which they call *Music Tag Conditioning*, and free-form text, called *End-to-End Text Conditioning*. To achieve this, they train two models with the two text formats and manually evaluate the text-music relevance of the generated music, concluding that the End-to-End Text Conditioning method obtains better text-music relevance than using the Music Tag Conditioning method. The authors consider that the main reason for this difference might be that the human-made music tag selection rules introduce much noise and result in the loss of some useful information from the original text.

Figure 4.4 – ERNIE-Music overall architecture.



Source: (Zhu et al., 2023)

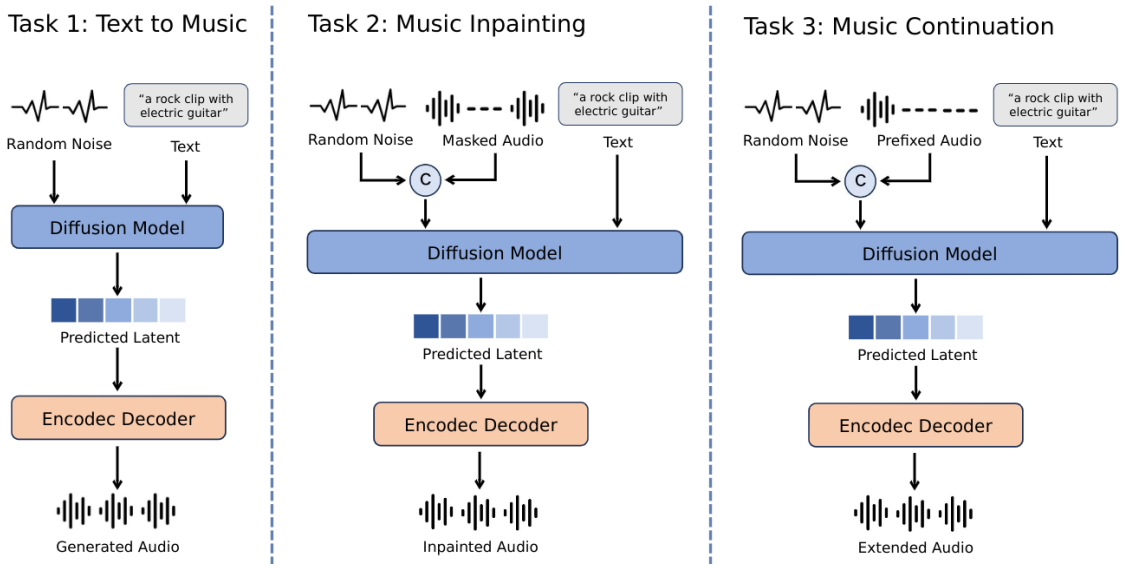
4.4 JEN-1: TEXT-GUIDED UNIVERSAL MUSIC GENERATION WITH OMNI-DIRECTIONAL DIFFUSION MODELS

The majority of the TTM models operates on spectogram representations of the audio, incurring fidelity loss from audio conversion. Others employ inefficient autoregressive generation or cascaded models, like MusicGen (Agostinelli et al., 2023), where

their training objectives are confined to single task, lacking the versatility of humans who can freely manipulate music. JEN-1 (Li et al., 2023a) represents a significant advancement in text-guided music generation, employing in-context learning and is trained with multi-task objectives, enabling music generation, music continuation, and music inpainting within a single model, this multi-tasking is illustrated in figure 4.5. To achieve this goal, the authors proposes a omnidirectional latent diffusion model, and additional masked music information, which the model is conditioned upon, can be extracted into latent embedding and stacked as additional channels in the input. The architecture of the omnidirectional diffusion model enables various input pathways, facilitating the integration of different types of data into the model, resulting in versatile and powerful capabilities for noise prediction and diffusion modeling. JEN-1 integrates the unidirectional diffusion mode by employing a unidirectional self-attention mask and a causal padding mode in convolutional blocks.

To avoid the spectrogram conversion losses, JEN-1 uses a masked autoencoder and diffusion model to directly model waveforms, which effectively reduces noises and mitigates artifacts, generating high-fidelity 48kHz stereo audio, and also integrates autoregressive and non-autoregressive diffusion to balance dependency modeling and generation efficiency.

Figure 4.5 – Illustration of the JEN-1 multi-task training strategy, including the text-guided music generation task, the music inpainting task, and the music continuation task. JEN-1 achieves the in-context learning task generalization by concatenating the noise and masked audio in a channel-wise manner. JEN-1 integrates both the bidirectional mode to gather comprehensive context and the unidirectional mode to capture sequential dependency.



Source: (Li et al., 2023a)

Table 4.1 – Models Information Summary

Feature	MusicLM	ERNIE-Music	MusicGen	JEN-1
Creators	Google	Baidu Inc.	Meta Research	Futureverse
Release Date	26 Jan 2023	9 Feb 2023	8 Jun 2023	9 Aug 2023
Architecture	Transformer	Diffusion	Transformer	Diffusion And Transformers
Sample Rate	24 kHz	16kHz	32 kHz	48kHz Stereo
Dataset	MusicCaps	Web Music w/ Text	MusicCaps	Private Music Data
Task	Single-Task	Single-Task	Single-Task	Multi-Task

Table 4.2 – Self-Reported Evaluation Metrics

Model	$\text{FAD}_{\text{VGG}} \downarrow$	$\text{FAD}_{\text{Trill}} \downarrow$	KL $\downarrow$	CLAP$\uparrow$
ERNIE-Music	-	-	-	-
MusicLM	4.0	0.44	1.01	-
MusicGen	3.8	-	1.22	0.31
JEN-1	2.0	-	1.29	0.33

4.5 Models Summary

In table 4.1, we have summarized the main features and characteristics of each model discussed in this section. This summary provides a quick reference to understand the distinct capabilities, methodologies, and applications of each model in the context of audio processing and music generation. In table 4.2, we display the metrics reported by the authors in each corresponding paper (Zhu et al., 2023; Agostinelli et al., 2023; Copet et al., 2023; Li et al., 2023a).

5 STATE-OF-THE-ART METRICS

5.1 CLAP

CLAP (Contrastive Language-Audio Pretraining) (Wu et al., 2023) is a pipeline of contrastive language-audio pretraining that considers different audio and text encoders, and, similar to the Contrastive Language-Image Pretraining (CLIP), where it learns the correspondence between text and image by projecting them into a shared latent space, CLAP aims to develop an audio representation by combining audio data with natural language descriptions based on their overlapping information with focus on the coherence between them. The overall architecture of clap can be viewed on figure 5.1

The authors publicly released LAION-Audio-630K, a large-scale audio-text dataset consisting of 633,526 pairs with a total duration of over 4,300 hours. In addition, the model also includes the feature fusion mechanism, designed to handle the variability in audio lengths. This mechanism allows for efficient processing of different lengths of audio inputs, combining global and local audio information, and significantly reducing computational inefficiency associated with long audio. Another key aspect of CLAP is its keyword-to-caption augmentation, which expands labels or tags of datasets like AudioSet into detailed captions. This expansion not only enriches the dataset but also contributes to more effective training of the contrastive language-audio pretraining model.

The paper conducts comprehensive experiments in text-to-audio retrieval, zero-shot audio classification, and supervised audio classification. These experiments demonstrate the model's superior performance in text-to-audio retrieval and its state-of-the-art results in zero-shot settings for audio classification tasks.

5.2 Fréchet Audio Distance

Traditional metrics like Signal to Distortion Ratio (SDR) and Signal to Interference Ratio (SIR) focus on how closely the enhanced audio matches the original studio recording, but they often fail to capture the perceptual quality of the music. This gap led to the development of the Fréchet Audio Distance (FAD), a metric specifically designed to evaluate the quality of generated audio clips. Based on the concept of the Fréchet Inception Distance (FID), widely used in the domain of image generation, FAD adapts this idea to the audio realm.

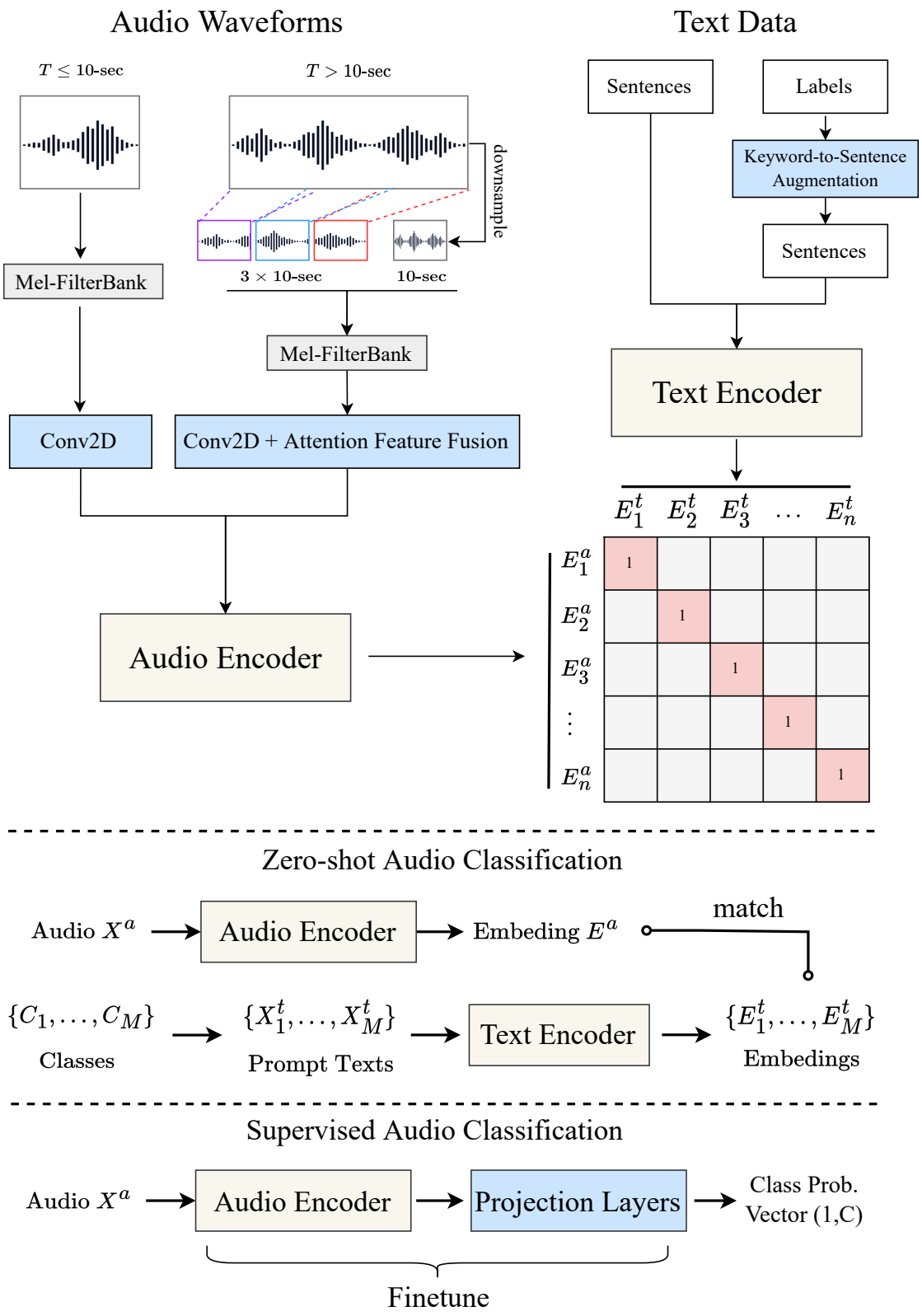
Unlike traditional metrics, the FAD score is computed by comparing an entire set of audio samples to a reference set in terms of their respective distributions in an embedding space as a whole, rather than individual clips. The process begins with the conversion of audio clips into embeddings, usually using the VGGish model, which encapsulate the key characteristics of the audio to be evaluated. The statistical distribution of these embeddings is then computed for both the real and generated datasets, as shown in figure 5.2. Given a multinormal fit to the distribution of audio embedding features with means μ_r and μ_t and covariance matrices Σ_r and Σ_t for the reference and test set, respectively, the FAD is

$$\text{FAD} = \|\mu_r - \mu_t\|^2 + tr\left(\Sigma_r + \Sigma_t - 2\sqrt{\Sigma_r \Sigma_t}\right), \tag{5.1}$$

where $tr(\cdot)$ is the matrix trace. A lower FAD score indicates that the generated audio is more similar to the real audio, suggesting higher quality and greater realism in the generated samples. In summary, FAD is a metric which is designed to measure how a given audio clip compares to clean, studio recorded music.

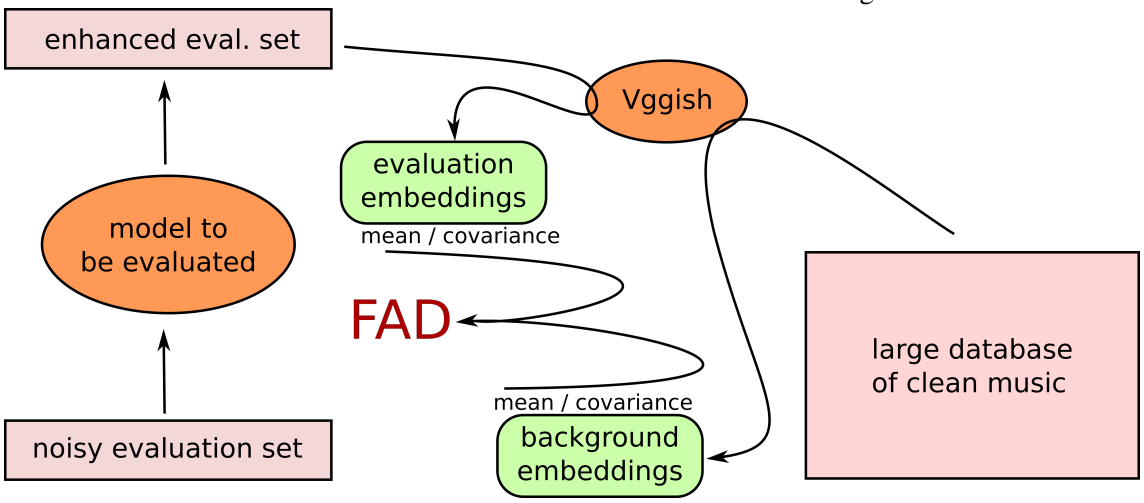
FAD has been widely adopted in the research community for evaluating various audio generation tasks as an objective music quality metric, including music synthesis, speech generation, and sound effect production. One of the key strengths of FAD is it's ability to evaluate audio quality without the need for human judgment, which can be subjective and inconsistent. This makes FAD a reliable and scalable tool for comparing different audio generation models, tracking improvements in model performance over time, and benchmarking against other methods in the field. However, while FAD is often used as a proxy for perceptual quality (Agostinelli et al., 2023; Copet et al., 2023; Li et al., 2023a), the underlying assumption given (5.1) is that the reference set is of high quality, that the audio embeddings capture features related to quality, that the embedding distribution can be approximated by a multinormal fit, and that the resulting single FAD score for the entire test set is a meaningful metric of model performance.

Figure 5.1 – The architecture of CLAP proposed model, including audio/text encoders, feature fusion, and keyword-to-caption augmentation.



Source: (Wu et al., 2023)

Figure 5.2 – FAD computation overview: using a pretrained audio classification model, VGGish, embeddings are extracted from both the output of a enhancement model that we wish to evaluate and a large database of background music. The Fréchet distance is then computed between multivariate Gaussians estimated on these embeddings.



Source: (Kilgour et al., 2019)

6 COMPARISON RESULTS

In this chapter, we evaluate the models performance using the previously mentioned metrics. For running the evaluation algorithms, some experiments in this work used the PCAD infrastructure[1], at INF/UFRGS.

6.1 FAD Score

In the literature, a commonly used reference set to calculate FAD is to only use the MusicCaps (Agostinelli et al., 2023) dataset as a reference set, which consists of 10 s music-related segments from Youtube videos, however, MusicCaps has a large portion of the samples labeled as "low-quality" by the expert raters. Furthermore, FAD is computed on audio embeddings, but embedding models trained with different architectures, losses, and data may capture different aspects of the audio. The great majority of TTM models published so far only generate the VGGish embedding for calculating the FAD score, lacking variability and potentially propagating the weaknesses of the VGGish model itself.

To further elaborate on the problems of using only this configuration, we use the research done by (Copet et al., 2023), where it asked human raters to evaluate the sound quality of the music generated by MusicGen using different parameters and configurations of the model, which are the model pre-trained with 1.5B parameters and the output with and without a melody to guide the generation, and also the output without melody where the model was trained with 3.3B parameters. Table 6.1 shows the objective and subjective metrics for the model MusicGen, where the objective metric is the FAD score calculated over the VGGish embedding and with the MusicCaps dataset as the baseline for the evaluation. We can see that even though the setting without a pre-defined melody and with 1.5B parameters on the model training showed the lowest FAD with a score of 3.4, the human raters evaluated it as the one with the lowest sound quality, proving that using the VGGish embedding with MusicCaps dataset as a baseline doesn't correlate properly with the human auditory perception.

To have a more robust comprehension of the generated output quality of the analysed models, it's necessary to consider different reference sets for comparison, as well as different embeddings to extract different features from the audio. Following (Gui

[1] gppd-hpc.inf.ufrgs.br (accessed in Dec 2023 and Jan 2024)

Table 6.1 – MusicGen generated music evaluation for FAD score and subjective music quality. Three different versions of the model are tested.

	MUSICCAPS Test Set	
MODEL	$\text{FAD}_{vgg} \downarrow$	OVL. $\uparrow$
MusicGen w.o melody (1.5B)	3.4	80.74$\pm$1.17
MusicGen w.o melody (3.3B)	3.8	**84.81**$\pm$0.95
MusicGen w. random melody (1.5B)	5.0	81.30$\pm$1.29

Hannes Gamper, 2024), we'll use the FMA-Pop (Defferrard et al., 2017) as a new reference set to calculate the FAD score, consisting of 4230 songs from the Free Music Archive (FMA). Beyond that, we also consider different embeddings to investigate how the different features that can be extracted from the same audio has an impact on the overall result and evaluation score. To address this, we use the Microsoft publicly available Frechet Audio Distance Toolkit[2] (Gui Hannes Gamper, 2024), computing the following embeddings of the generated music of the models:

CLAP (Microsoft). Microsoft's Contrastive Language-Audio Pretraining (CLAP) [3] (Elizalde; Deshmukh; Wang, 2023) is trained with 128,000 audio and text pairs, has shown state-of-the-art performance in Zero-Shot learning tasks across multiple domains including sound event classification, scenes, music, and speech. We describe how CLAP works in section 5.1

CLAP-Music (LAION). For LAION's CLAP [4](Wu et al., 2023) in music, the pretrained model used is "music audioset epoch 15 esc 90.14" (L-CLAP mus). This model is specifically trained and tailored for music-related tasks, leveraging LAION's large-scale dataset and expertise in audio processing.

CLAP-Audio (LAION). The LAION CLAP (Wu et al., 2023) model for general audio, labeled as "630k-audioset-best" (L-CLAP aud), is designed to handle a broader range of audio types. This model is expected to have a wide application in various audio processing tasks beyond music, including general sound events and scenes.

Encodec. The EnCodec model [5] (Défossez et al., 2022b) encoder-decoder architecture is described in section 3.1. This version of EnCodec is a causal model operating at 24 kHz on monophonic audio trained on a variety of audio data. It can compress audio to 1.5, 3,

[2] github.com/microsoft/fadtk (accessed in Dec 2023 and Jan 2024)
[3] github.com/microsoft/CLAP (accessed in Dec 2023 and Jan 2024)
[4] github.com/LAION-AI/CLAP (accessed in Dec 2023 and Jan 2024)
[5] github.com/facebookresearch/encodec (accessed in Dec 2023 and Jan 2024)

6, 12 or 24 kbps.

Encodec (48k). This version of EnCodec is a non-causal optimized for 48 kHz stereophonic audio trained on music-only data. The EnCodec (48k) model retains the core characteristics of the standard EnCodec but is tailored for higher fidelity in stereo audio settings.

MERT-v1-95M. In section 3.2, we explained how MERT functions. In this version of MERT[6] (Li et al., 2023b), the number of parameters that are loaded to memory is 95 million. The main differences from the first version this model (MERT-v0) is that v1 implements a MLM prediction with in-batch noise mixture, is trained with higher audio frequency (24K Hz) and more audio data (up to 160 thousands of hours). The number of transformer layers and the corresponding feature dimensions also can be outputted from the model. This is marked out because features extracted by different layers could have various performance depending on tasks.

VGGish. VGGish [7], developed by Google, is a well-known audio feature extraction model and the most commonly used for measuring the FAD Score. It's based on the VGG architecture (Simonyan; Zisserman, 2015), commonly used in image processing, and adapted for audio. VGGish processes audio into log-Mel spectrogram patches and outputs 128-dimensional embeddings. It's pre-trained on a large-scale YouTube dataset and is often used as a feature extractor for various audio classification tasks.

DAC-44kHz. The standard model of Descript Audio Codec[8] (Kumar et al., 2023) discussed on section 3.5, where it compresses 44.1 KHz audio into discrete codes at 8 kbps bitrate. Can be used on all domains (speech, environment, music, etc.), making it widely applicable to generative modeling of all audio.

CDPAM (Acoustic). As we describe in section 3.4, this corresponds to the acoustic embedding generated in the CDPAM (Manocha et al., 2021) model. Following the instructions of the authors, we convert the audio samples to 16-bit PCM audio files to perform correctly.

CDPAM (Content). Similarly to CDPAM Acoustic, this represents the content embedding generated by the CDPAM model. We also convert the audio samples to 16-bit PCM audio files.

Table 1 presents the comparison of the proposed evaluation method of the FAD

[6] huggingface.co/m-a-p/MERT-v1-95M (accessed in Dec 2023 and Jan 2024)

[7] github.com/tensorflow/models/tree/master/research/audioset/vggish (accessed in Dec 2023 and Jan 2024)

[8] github.com/descriptinc/descript-audio-codec (accessed in Dec 2023 and Jan 2024)

Table 6.2 – Multiple embeddings used for the calculation of the FAD Score evaluation for the proposed models.

MODEL	ERNIE-MUSIC	MUSICLM	MUSICGEN	JEN-1
CLAP MICROSOFT	548.9	254.3	342.1	259.9
CLAP-LAION-MUSIC	0.82	0.435	0.452	0.298
CLAP-LAION-AUDIO	0.669	0.279	0.428	0.419
ENCODEC	146.3	15.978	29.48	12.96
ENCODEC-48K	68.80	10.36	11.35	10.43
MERT-V1-95M	18.98	11.74	12.87	12.04
VGGISH	8.578	3.353	4.305	4.587
DAC-44KHZ	967.1	778.4	347.2	1183.8
CDPAM-ACOUSTIC	0.334	0.052	0.075	0.124
CDPAM-CONTENT	0.295	0.049	0.054	0.124

Score considering all the embeddings listed above for ERNIE-Music (Zhu et al., 2023), MusicLM (Agostinelli et al., 2023), MusicGen (Copet et al., 2023) and JEN-1 (Li et al., 2023a). As there is no official public implementation for ERNIE-Music[9], MusicLM[10] and JEN-1[11], we use these models public demos for our tests. For comparing these results with the subjective overall quality rated by humans, we report the research done by (Li et al., 2023a) for the models MusicLM, MusicGen, and Jen-1, where they asked human raters to evaluate the audio quality of the generated samples of these models from a score in the range of 1 to 100, where 100 would be the highest quality, and the results were 81.7, 83.8 and 85.7 respectively. We can see that, compared to the VGGish embedding, which is the standard embedding used in evaluations of the FAD score, the MERT-v1-95M and EnCodec-48k present results more aligned with the human perception, and both of them are designed and trained on music datasets.

6.2 CLAP Score

To calculate the CLAP score discussed in section 5.1, we'll use the official Microsoft's implementation of the Contrastive Language-Audio Pretraining [12]. We use the 2023 pre-trained model to compute the CLAP scores of each song generated by the TTM models with it's corresponding text description, and report the mean of this output.

[9] ERNIE-Music-Generated-Cases (accessed in Dec 2022 and Jan 2023)
[10] google-research.github.io/seanet/musiclm/examples/ (accessed in Dec 2023 and Jan 2024)
[11] www.futureverse.com/research/jen/demos/jen1 (accessed in Dec 2023 and Jan 2024)
[12] github.com/microsoft/CLAP (accessed in Jan 2024 and Feb 2024)

Table 6.3 – Comparison of the CLAP score and overall subjective music alignment with state-of-the-art text-to-music generation models

	QUANTITATIVE	QUALITATIVE
MODEL	CLAP↑	ALI. ↑
ERNIE-Music	13.65	-
MusicLM	16.35	82.0
MusicGen	15.79	79.5
JEN-1	11.76	82.8

To evaluate the alignment of Microsoft's CLAP score and the human perception, we also use Jen-1 (Li et al., 2023a) research conducted on human raters that subjectively measures the coherence between the input text and generated music of TTM models. We report in table 6.3 the score of MusicGen, MusicLM and Jen-1 subjectively alignment and the score of our evaluation on Microsoft's CLAP for all of the 4 models. This analysis juxtaposes the CLAP scores against subjective evaluations of coherence between the generated music and input text, revealing a divergence between objective scores and human perception. Notably, the CLAP score's divergence between MusicLM and JEN-1, despite similar subjective evaluations, highlights the metric's sensitivity to nuances not captured in human ratings. This observation prompts a deeper inquiry into the alignment of objective metrics like CLAP with subjective human judgment, potentially guiding future refinements in evaluative methodologies for music generation models.

7 CONCLUSION

This work has explored the domain of music generation using text-to-music (TTM) models, offering a comprehensive analysis of the latest methodologies, architectures, and evaluation metrics in the field. Through an extensive review of the models MusicLM, ERNIE-Music, MusicGen, and JEN-1, this research has highlighted the significant advancements in generating music that aligns with textual descriptions, demonstrating the potential of these models to revolutionize how we interact with music creation and understanding.

The evaluation of these models using state-of-the-art metrics such as FAD and CLAP scores revealed insightful findings on their performance, underscoring the importance of robust, multifaceted evaluation frameworks to accurately capture the nuances of generated music quality and text-audio coherence. This work has shown that while objective metrics provide essential insights into model performance, they also contain many discrepancies between these objective metrics and human perception, evident in the divergent CLAP and FAD scores of the models that otherwise performed similarly in the reported subjective evaluations.

Moreover, the exploration into different embeddings and reference sets for FAD calculation has underscored the necessity of diversifying evaluation methodologies to ensure a comprehensive and nuanced assessment of music generation models. By considering various aspects of audio quality and textual coherence, this research advocates for a more holistic approach to evaluating TTM models, one that mirrors the complexity and richness of human musical experience.

7.1 Limitations

This research, while comprehensive in its approach to evaluating text-to-music generation models, encounters several limitations that are important to acknowledge. Firstly, the analysis was partly constrained by only having access to the demos of some music generation models. This limited access restricted the ability to conduct a more robust, in-depth evaluation and comparison across a full spectrum of model outputs, potentially affecting the insights into each model's capabilities and performance nuances.

Additionally, the study did not incorporate a qualitative research component involving human raters, instead, we use the qualitative research done by other authors

(Copet et al., 2023; Li et al., 2023a). The absence of subjective evaluations from human listeners represents a significant limitation, as it precludes a holistic understanding of the generated music's perceptual and emotional impact. Human evaluation is crucial for assessing aspects of musical quality, such as emotional expression, overall audio quality, and the coherence between the generated music and the provided textual descriptions, which current objective metrics might not fully capture.

7.2 Future Work

In light of the limitations identified, new possibilities for future work emerge, promising to extend the research's scope and depth in evaluating text-to-music generation models. An immediate area for expansion involves utilizing a larger dataset of outputs from the generated models. By analyzing a more extensive and varied collection of music samples, future research can offer a more robust and less noisy evaluation of a model's performance, encompassing a wider array of musical genres, styles, and expressions.

Furthermore, the study could benefit from analyzing additional state-of-the-art models not covered in this work. Given the rapid pace of advancements in the field, examining newer models would provide valuable insights into evolving methodologies, architectures, and capabilities, enhancing our understanding of the current landscape of music generation technologies.

Lastly, future research should consider evaluating the models using additional objective metrics, like the Kullback–Leibler Divergence (KLD) and MuLan Cycle Consistency (MCC), alongside more implementations of the CLAP score. Employing these metrics would offer a different perspective on the distributional characteristics of the generated music compared to reference datasets, potentially uncovering new information and a more in-depth knowledge of the models performance. Similarly, exploring diverse implementations of the CLAP score could refine the assessment of textual-audio coherence, providing a more robust evaluation of how effectively models translate textual descriptions into musically expressive outputs, and also detect potential flaws within the CLAP implementations already created.

REFERENCES

AGOSTINELLI, A. et al. Musiclm: Generating music from text. **arXiv preprint arXiv:2301.11325**, 2023.

BALESTRIERO, R. et al. A cookbook of self-supervised learning. **arXiv preprint arXiv:2304.12210**, 2023.

BORSOS, Z. et al. **AudioLM: a Language Modeling Approach to Audio Generation**. 2023. Available from Internet: <https://arxiv.org/abs/2209.03143>.

BROWN, J. C. Calculation of a constant q spectral transform. **The Journal of the Acoustical Society of America**, v. 89, n. 1, p. 425–434, 1991.

CHUNG, H. W. et al. Scaling instruction-finetuned language models. **arXiv preprint arXiv:2210.11416**, 2022.

CHUNG, Y.-A. et al. W2v-bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training. **arXiv preprint arXiv:2108.06209**, 2021.

COPET, J. et al. Simple and controllable music generation. **arXiv preprint arXiv:2306.05284**, 2023.

DEFFERRARD, M. et al. FMA: A dataset for music analysis. 2017. Available from Internet: <https://arxiv.org/abs/1612.01840>.

DéFOSSEZ, A. et al. High fidelity neural audio compression. **arXiv preprint arXiv:2210.13438**, 2022.

DéFOSSEZ, A. et al. High fidelity neural audio compression. **arXiv preprint arXiv:2210.13438**, 2022.

ELIZALDE, B.; DESHMUKH, S.; WANG, H. **Natural Language Supervision for General-Purpose Audio Representations**. 2023. Available from Internet: <https://arxiv.org/abs/2309.05767>.

GERMAIN, F. G.; CHEN, Q.; KOLTUN, V. Speech denoising with deep feature losses. 2018. Available from Internet: <https://arxiv.org/abs/1806.10522>.

GOODFELLOW, I. J. et al. Generative adversarial networks. 2014. Available from Internet: <https://arxiv.org/abs/1406.2661>.

GUI HANNES GAMPER, S. B. D. E. A. Adapting frechet audio distance for generative music evaluation. 2024. Available from Internet: <https://arxiv.org/abs/2311.01616>.

HO, J.; JAIN, A.; ABBEEL, P. Denoising diffusion probabilistic models. **arXiv preprint arXiv:2006.11239**, 2020.

HUANG, Q. et al. Mulan: A joint embedding of music audio and natural language. **arXiv preprint arXiv:2208.12415**, 2022.

HUANG, R. et al. Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models. **arXiv preprint arXiv:2301.12661**, 2023.

KILGOUR, K. et al. Fréchet audio distance: A metric for evaluating music enhancement algorithms. **arXiv preprint arXiv:1812.08466**, 2019.

KONG, J.; KIM, J.; BAE, J. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. **Advances in Neural Information Processing Systems**, v. 33, p. 17022–17033, 2020.

KONG, Z. et al. Diffwave: A versatile diffusion model for audio synthesis. **arXiv preprint arXiv:2009.09761**, 2020.

KREUK, F. et al. Audiogen: Textually guided audio generation. **arXiv preprint arXiv:2209.15352**, 2023.

KUMAR, R. et al. High-fidelity audio compression with improved rvqgan. 2023. Available from Internet: <https://arxiv.org/abs/2306.06546>.

LI, P. et al. Jen-1: Text-guided universal music generation with omnidirectional diffusion models. **arXiv preprint arXiv:2308.04729**, 2023.

LI, Y. et al. Mert: Acoustic music understanding model with large-scale self-supervised training. 2023. Available from Internet: <https://arxiv.org/abs/2306.00107>.

LIU, H. et al. Audioldm: Text-to-audio generation with latent diffusion models. **arXiv preprint arXiv:2301.12503**, 2023.

MANOCHA, P. et al. A differentiable perceptual audio metric learned from just noticeable differences. 2020. Available from Internet: <https://arxiv.org/abs/2001.04460>.

MANOCHA, P. et al. Cdpam: Contrastive learning for perceptual audio similarity. 2021. Available from Internet: <https://arxiv.org/abs/2102.05109>.

OORD, A. van den; VINYALS, O.; KAVUKCUOGLU, K. Neural discrete representation learning. 2018. Available from Internet: <https://arxiv.org/abs/1711.00937>.

RAFFEL, C. et al. Exploring the limits of transfer learning with a unified text-to-text transformer. **arXiv preprint arXiv:1910.10683**, 2020.

RAMESH, A. et al. **Zero-Shot Text-to-Image Generation**. 2021. Available from Internet: <https://arxiv.org/abs/2102.12092>.

RUBENSTEIN, P. K. et al. Audiopalm: A large language model that can speak and listen. 2023. Available from Internet: <https://arxiv.org/abs/2306.12925>.

SHEFFER, R.; ADI, Y. I hear your true colors: Image guided audio generation. **arXiv preprint arXiv:2211.03089**, 2023.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. 2015. Available from Internet: <https://arxiv.org/abs/1409.1556>.

SOHL-DICKSTEIN, J. et al. Deep unsupervised learning using nonequilibrium thermodynamics. In: **International Conference on Machine Learning**. [S.l.: s.n.], 2015. p. 2256–2265.

TAN, X. et al. A survey on neural speech synthesis. **arXiv preprint arXiv:2106.15561**, 2021.

TOUVRON, H. et al. Llama: Open and efficient foundation language models. **arXiv preprint arXiv:2302.13971**, 2023.

VASWANI, A. et al. Attention is all you need. **arXiv preprint arXiv:1706.03762**, 2017.

VASWANI, A. et al. Attention is all you need. 2023. Available from Internet: <https://arxiv.org/abs/1706.03762>.

WU, Y. et al. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. **arXiv preprint arXiv:2211.06687**, 2023.

YANG, D. et al. Diffsound: Discrete diffusion model for text-to-sound generation. **arXiv preprint arXiv:2207.09983**, 2023.

ZEGHIDOUR, N. et al. Soundstream: An end-to-end neural audio codec. **arXiv preprint arXiv:2107.03312**, 2021.

ZHU, P. et al. Ernie-music: Text-to-waveform music generation with diffusion models. **arXiv preprint arXiv:2302.04456**, 2023.

ZIYIN, L.; HARTWIG, T.; UEDA, M. Neural networks fail to learn periodic functions and how to fix it. 2020. Available from Internet: <https://arxiv.org/abs/2006.08195>.