Proposal - Master Thesis - Artificial Intelligence

Rhythm and Reason: Adding fine-grained control to deep learning music generation models.

First examiner:

Anja Volk

Second examiner:

Peter van Kranenburg

Candidate:

Efraim Dahl

February 21, 2025

Abstract

Music is essential in video games, enhancing immersion and engagement, but players may disengage due to excessive repetition or personal preferences. This is particularly problematic in games like LastMinuteGig, a Musical Attention Control Training (MACT) application for Parkinson's patients. To maintain engagement and the effect of the intervention, a diverse set of controlled, adaptive music is needed. Advances in generative music offer a scalable solution. This research explores efficient methods of adding control to pre-trained models, to steer towards rhythmically adaptive music suitable for MACT. Insights from small-scale experiments will inform the application of rhythmic control in MusicLang, a transformer based music generator, which will be used to build musical assets for LastMinuteGig.

Contents

1. Introduction

Music is a cornerstone in modern video games. Music evokes emotional responses, triggers memories, can direct attention and improve immersion and the overall experience of a game. Yet, it is not uncommon for players to become disengaged from the game music, often due to excessive repetition or simply different personal preferences.Rogers and Weber, 2019 This can be a problem in serious games for therapeutic use such as "Last Minute Gig" a gamified Musical Attention Control Training (MACT) application, aimed to improve attention control in patients with Parkinson's disease Chalkiadakis, 2022. Ideally MACT is a personalized experience with dynamic adjustments taking into account a patient's abilities and preferences. In the context of gamified self-directed therapy,this is crucial to keep patients engaged and progressing.[1] In order to accommodate a wide array of patient's abilities and preferences in a MACT-game, a considerable amount of musical material, specifically music that fits the constraints of MACT is required. One potential avenue to provide this is through controlled music generation.

Over the last years, music generation has grown from a mostly academic endeavor to a billion-dollar industry. Large technology companies are exploring music generation with foundation models such as Meta's MusicGenCopet et al., 2023 or Google's MusicLM Agostinelli et al., 2023. The commercial startup Suno is valued at 500 million dollars, while chart-topping songs are being generated with Udio Ferdinand Meyen, 2024Stassen, 2024. This breakthrough is powered mostly by advances in language modelling, bringing about Large Language Models such as GPT-4 or LLAMA3 applied to music.

In the context of games, generative music is promising to cater to different user preferences and enable a richer, more varied musical experience than conventional aproaches to adaptive audio. Video games can include hundreds of hours of game-play and branching story lines. When additional complexity is added to adapt the music to individual preferences, it quickly is no longer feasible to manually compose unique music to fit all scenarios. Especially in more resource constrained applications such as games for therapeutic use, generated music can offer an avenue to provide a large variety of music that can improve user engagement, and ultimately the effect of the intervention. One of the key components of a successful generative model is control. In music generation control could be a text-description, information about

---

[1]This type of therapy is not aimed at replacing traditional guided therapy. Ideally it is used as supplement, and may provide relief in situations where traditional therapy is not immediatly available

genre, style or instrumentation. Control can also include fine-grained musical parameters such as a melody, chord progression or musical structure. Generative models can often be controlled with human-composed music, where the provided music is continued, interpolated between or inpainted (generating accompaniments, melodies, or additional musical lines). In LastMinuteGigChalkiadakis, 2022 the music periodically provides stimuli, a noticeable change in the music, based on which the user changes their playing, specifically their rhythm. One way this stimulus could be achieved in generated music is by adding control that introduces a shift in rhythmic patterns at semi regular intervals.

There are a variety of ways to achieve control in a music generator. For rule based systems, control is integrated explicitly. In statistical systems (including deep learning) control is typically applied through architectural constraints and choice of training data. Certain deep learning architectures lend themselves to certain types of control. I.e an auto-regressive transformer models a sequence based on prior parts of the sequence, it natively generates continuations of an input. In deep learning, other musical parameters (i.e chord progression or melody) can be controlled for by joint conditioning of a model, which means that the controlled musical parameters are made explicit while training. This method of adding control is quite cost-intensive, requiring as much, or even more training material as the baseline model without controls.

A more economical method of adding control to a deep learning generative model is through fine-tuning and post-hoc conditioning. In fine-tuning (sometimes referred to as transfer learning) a pre-trained model is trained on additional data. There are various methods of fine-tuning, involving different configurations of the model, the model parameter and the integration of control input, but typically fine-tuning uses considerably fewer resources and data than training a model from scratch. In post-hoc conditioning a generative model is adjusted while it is running to increase the likelihood of the output to fit the desired constraints. The goal of this thesis is to steer a pretrained generative model to generate music fit to an MACT context, without training a large model from scratch, but using efficient fine tuning to introduce new control mechanisms to a pretrained model. For this I will first explore and evaluate different methods of adding control using small generative music model. I will then apply the most successful techniques of adding learned from this first step to a larger generative music model, MusicLang to steer the output towards something usable in the context of MACT.

In a final step we will generate a large amount of music to extend the LastMinuteGig video game for MACT in patients with Parkinson's. LastMinutGig is a mobile application built with Unity3D. In its current iteration it contains a single button with which a player taps along to the music, controlling a set of guitar chords. When a musical change occurs the player is instructed to change their tapping. It uses a simple music generation process based on recombination,

which I aim to replace with a set of generated music.

2. Literature

The following section contains an extensive (though not exhaustive) categorical literature study. The first part provides contextual information, including motivations, and ethical concerns as well as a comprehensive overview of techniques used in music generation. The second half is more technical, with a discussion on music representation, tokenization, control and fine tuning. As a starting point, I used the most recent ISMIR (International Society for Music Information Retrieval) papers on music generation and their referenced sources, alongside other papers suggested by my advisor, Anja Volk. In addition, I performed systematic searches using the following keywords: Deep Learning Music Generation, Diffusion Music Generation, Transformer Music Generation, Symbolic Diffusion Music Generation, and Control Music Generation.

2.1 Why Generate Music? - Motivation

2.1.1 Composition Co-Pilots

The earliest music generators (including the 18th-century musical dice gamesNierhaus, 2009) were justified as methods to inspire composers and music makers, including enabling novice composers to write music. Composer David CopeCope, 1989 states that he turned to music generation to overcome writer's block. Commercial enterprises like Suno claim to be "building a future where everybody can make great music" n.d. Many current and past efforts highlight music generation as a process that assists composers. In DeepBachHadjeres et al., 2017, the authors go to great lengths to make the system flexible and usable in real-life composition scenarios. Initially, they developed a MuseScore integration and later tied DeepBach into the web app NONOTO[1] to support musical inpainting in Ableton Live scores. Similarly, with Composer's AssistantMalandro, 2023, the authors explicitly enable musical inpainting and continuation in the REAPER music production software. AI has also been explored as a co-improviser for live music-making, such as in Pachet's ContinuatorPachet, 2003 and Ben-Tal's musical dialog systemKite-Powell, 2023.

2.1.2 Music in Games

Beyond co-creating music, AI-generated music serves functional purposes, such as background music in videos and therapy-assisting games. In gaming, AI-generated or AI-assisted music is

---

[1]https://github.com/SonyCSLParis/music-inpainting-ts

particularly relevant due to the scale and interactivity of games.

Video games often facilitate hundreds of hours of gameplay, featuring branching storylines and complex player interactions. However, game soundtracks typically cover only a fraction of that timePlut and Pasquier, 2020; Worrall and Collins, 2024. Through different adaptive techniques, relatively short snippets of original material can be stretched into hours of unique audio, often relying on the recombination of different elements. However, rule-based recombination has its limitations. A recent study of player behavior Rogers and Weber, 2019 finds that many players eventually turn off game music. They cite various reasons, such as preferring their own music over the game soundtrack (46.7%) or finding the in-game music repetitive (29.6%).

Procedural generation of 3D assets, levels, and enemy behavior is commonplace, but music generation remains underutilized. Composing and adapting music for every possible scenario would be tedious. AI-assisted music composition could enable adaptive audio on a large scale, either by creating numerous musical assets anticipating player choices or generating new variations of the game score in real time to enhance player immersion. However, several challenges make AI music generation difficult in games. Performance issues arise due to the resource-intensive nature of AI generators. Additionally, AI-generated music is difficult to control, and there is little guarantee that the generated tracks will be appropriatePlut and Pasquier, 2020. Furthermore, AI music generators currently lack proper integration into video game environments and enginesWorrall and Collins, 2024.

2.1.3 Music in Serious Games

Aside from games for general audiences, generative music has potential applications in serious games. Serious games are designed for purposes beyond entertainment, such as education or therapeutic use, including music therapyDjaouti et al., 2011. In music therapy, music can be used for emotion regulation, motivation, adherence, motor coordination, rhythmic entrainment, and facilitating social interactionsAgres et al., 2021. Musical attention control training has been shown to help individuals with Parkinson'sPark and Kim, 2021, ADHDMartin-Moratinos et al., 2023, autismPasiali et al., 2014, and psychosisvan Alphen et al., 2019 improve their mental capabilities for selective and switching attention. Serious games have the potential to supplement music therapy. "Last Minute Gig"Chalkiadakis, 2022 implements clinical music therapy protocols as a serious game to improve attention control in Parkinson's patients. However, users reported boredom and a lack of feedback, while more musically experienced users felt less challenged. SchletteSchlette, 2022 attempted to address these issues by introducing dynamic difficulty adjustment through a feedback system alongside more complex music generation. This thesis aims to develop a controlled music generation model to improve player

engagement through a richer music system.

2.2 Why Not Generate Music? - Ethical Concerns

2.2.1 Introduction

There are several concerns related to AI-based music generation. First, there are legal concerns regarding copyright and licensing. Generative models may produce outputs that are identical or highly similar to copyrighted training data. Furthermore, the question remains whether models should be allowed to train on copyrighted data in the first place. Broader concerns include the devaluation of human labor and creativity, the oversaturation of cultural spaces with low-quality generated content, and the environmental impact of large generative models, which require substantial energy, water, and rare materials to operate.

2.2.2 Data Leakage and Copyright

Generative AI companies are achieving record-breaking valuations and this includes music generators with Suno at a valuation of about 500 million dollars after a 125 million dollar fund-raiser leading the pack. Stassen, 2024 Tencer, 2024. However, AI companies are facing backlash from artists and record labels with an organization of record labels including the "big three" Sony, Warner Music, and UMG suing Suno and Udio for $150.000 dollars per infringed workKaba et al., 2024. Generative AI runs a substantial risk of parroting or leaking training data. In language models, the leakage problem is of concern when training on data that contains sensitive information, which may be revealed either through accidental leakage or through membership inference attacks Duan et al., 2024. While leakage may raise privacy concerns in other generative models such as speech and image-generators Carlini et al., 2023 for music, the risk of training data leakage is mostly an issue of copyright. Ed Newton Rex shows some examples of how Suno can be influenced Newton-Rex, 2024 to leak training data. This ability to create disconcertingly close reproduction of copyrighted work is also cited in the court documents. Suno has since started to prevent prompting with artist names (i.e. in the style of Eminem) and including known song lyrics.

2.2.3 Training and Copyright

Besides leaking training data, there is the more general question of whether AI models should be allowed to train on unlicensed work. Echoing the court case between OpenAI and the New York Times Reed, 2024, both Suno and Udio cite fair use in response to accusations of copyright infringement. In US copyright law the fair-use clause limits exclusive rights to a work, with four factors to consider: 1) purpose and character of the work in use, 2) nature of copyrighted

work, 3) amount of the copyrighted work used, and 4) the effect on the potential market or value of copyrighted work."Copyright Law of the United States Title 17, Chapter 1, Section 107", n.d. Fair use is often granted to derivative works such as parodies and covers and works used in educational settings. AI's learning of structures has also been likened to the human learning process, humans learn based on copyrighted music they listen to, without giving credit or compensation to their influences. Newton-Rex Newton-Rex, 2024 rejects this comparison. In learning music, humans contribute to the musical ecosystem, they take lessons, go to performances, or at the very least generate some streaming revenue for artists, none of these are true for machine learning models learning from scraped data.

2.2.4 Devaluing Music

While few are following director Ram Gopal Varma's announcement to only use AI-generated music in his future films Singh, 2024, AI-generated music is becoming increasingly difficult to differentiate from human production and already receiving considerable amounts of streams and it is not unlikely that music in film, video and game projects may be replaced or at least supplemented with AI-generated music. The online music market is saturated, with more than 100,000 songs uploaded to music streaming giant Spotify every day Stassen, 2023. Generative AI may just further exacerbate this problem, resulting in a race to the bottom for creatives and musicians.

2.2.5 Environmental Impact

Digital infrastructure, traditional data centers, crypto-currency mining, and AI-centered data infrastructure account for about 2% of the world's energy consumption Marechal, 2024. The performance of current large language models often scales with simultaneous increases in model size, training data, and computation time.Kaplan et al., 2020 Each of these three factors requires considerable resources. Music generation is no exception. In a recent ISMIR publication Holzapfel et al., 2024 the authors make estimates on energy consumption of different projects related to music generation and computation-intensive MIR, finding an average energy consumption of 224.8kWh for model training (the energy consumption of an average western person over 2 months). The energy consumption is divided highly unevenly, with the median being at merely 18 kwh (3 days of an average westerner's energy consumption). Music generation models associated with large technology companies are responsible for about 89% of the estimated energy use. This is only for training, models that are deployed publicly, continue to use substantial energy for inference. Beyond just the carbon footprint of generative AI, the local impacts of resource use such as rare minerals and water are important to keep in mind.

2.2.6   How I plan to address these concerns

During this thesis process, I'm planning to address the outlined ethical concerns in the following manner. First, I will only train, and use open-source models trained on licensed data providing attributions where possible. I will also make my trained models publicly available with substantial documentation to contribute to the open-source ecosystem, research, and music. Second I am explicitly designing the generative process to be cooperative, this is facilitated on the one hand through the introduction of new control modalities, and on the other the choice of symbolic music over audio, which eases editing and integration into composition software. Finally, I am not attempting to train custom large foundation models, rather I'm trying to find ways to extend existing models to introduce new control mechanisms, requiring training on only a fraction of the model parameters, with substantially less need for data, computation, and energy.

2.3   Overview of music generation

From antique wind-chimes to classical period musical dice games to the aleatoric music of the 20th century - humans have used algorithmic, probabilistic, and statistical methods to create music. As early as the late 1940s, computers have played a role in composition as sound generators, instruments France-Presse, 2016 and providing musical material themselves, such as in the 1957 Illiac suite Hiller and Isaacson, 1959. The following 50 years are characterized by disparate, academic experiments in music generation, mostly in the symbolic music domain. They utilize a variety of contemporary AI technologies, from expert systems and ontologies Hiller and Isaacson, 1959Ebcioğlu, 1994 to evolutionary algorithms Polito et al., 1997 to feed-forward Todd, 1989, recursive Mozer, 1994, and convolutional neural networks Chakraborti et al., 2020. Some composers in the classical tradition, such as Iannis Xenakis Xenakis, 1992 and David CopeCope, 1989 use computer algorithms in their creative work. In the 2010s a small ecosystem of commercial generative music startups such as Jukedeck, PopGun, and Ampermusic Featherstone, 2017 starts to emerge, alongside an increasing number of publications applying deep learning, particularly Generative Adversarial Networks (GANs), and Recursive Neural Networks (RNN) to music including MIDInet Yang et al., 2017, DeepBach Hadjeres et al., 2017 and FolkRNN Sturm and Ben-Tal, 2016. With the development of the transformer architecture in 2017 Vaswani et al., 2017, large technology companies start experimenting with music generators including OpenAI's Jukebox Dhariwal et al., 2020 and MusenetChristine, 2019, Meta's MusicGenCopet et al., 2023 and Google's MusicLM Agostinelli et al., 2023 and the preceding Magenta project, with fully generative models capable of producing sequences of high-quality music modeled from raw audio. At the time of writing, commercial music generators such as Suno and Udio are raising millions of dollars in investment fundsStassen, 2024, while gener-

ated music is being widely streamed and actively used in TV and video productions.

2.4 Non-neural music generation

2.4.1 Why look beyond deep learning

Neural, specifically deep learning (DL) systems, currently receive considerable attention. However, given their substantial drawbacks relating to explainability, transparency, computational efficiency, copyright and licensing issues, and their enormous need for data (more in section **??**), it is worthwhile considering alternative approaches to music generation. A recent study Yin et al., 2023 performs a comprehensive listening survey comparing neural net and non-neural net systems. The top-performing systems a Markov Model - MAIA Markov Collins and Laney, 2017 and a deep learning system - MusicTransformer C.-Z. A. Huang et al., 2018 perform similarly well in the listening study. The choice of one of the earliest transformer-based music from 2018 C.-Z. A. Huang et al., 2018 and the restriction to symbolic music, raises questions, whether their conclusion of similar performance still holds. Considering the study was published in 2023, these are important limitations. However, their criticism that many DL-based music generation projects do not look beyond DL and compare their systems based on technical metrics - with no obvious impact on how human listeners perceive the output remains solid. Finally, many hybrid approaches successfully combine traditional rule-based or statistical methods with deep learning. Those methods may help researchers and developers maintain a more comprehensive toolkit of techniques and paradigms.

2.4.2 Rule-based music generation

Non-neural net systems can classified as either rule-based or statistical. Both types have been part of some of the earliest attempts at music generation. Centuries of style-defining musicological writing from ancient Mesopotamian tuning charts Mirelman, 2013 to Fux's Gradus ad Parnassum Fux, 1725 and Arnold Schönberg's 12-tone music have crystallized sets of rules that approximate various styles of music. Many approaches to music generation take advantage of this knowledge and codify it into a computer program, creating expert systems for music generation. In Hiller & Isaacson's Illiac Suite, a series of experimental, computational compositions, the first and second movements are generated following the rules of first species counterpoint Fux, 1725, approximating Palestrina's contrapuntal technique. Some rules aim to contain the melody, such as limiting the range to an octave, enforcing the identical start and end notes, and avoiding consecutive melodic jumps. Other rules aim to constrain harmony, such as forbidding parallel octave, fifth, and fourth motion and enforcing consonant harmonies. Hiller and Issacson's approach is relatively simple, using only a handful of conditions (see figure **??**).

Rule-based generation can be highly complex, such as CHORAL Ebcioğlu, 1994 (for which the developer also built a custom programming language), which encodes over 300 rules to realize bach-style chorales from a given melody.
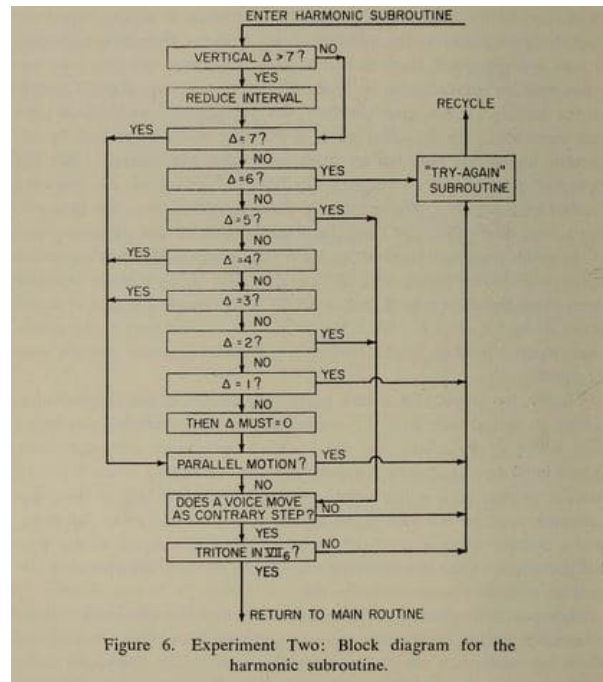

Figure 2.1: Rule-Based - Block Diagram from Hiller and Issacson's book - explaining movement two of the 1957 Illiac Suite.

2.4.3 Markov Model Music Generation

Markov models remain a popular method of generating music to this day. At its simplest, a Markov music generator could work off of a transition matrix for pitch classes, such as Richard Pinkerton's 1956 "Banal Melody Generator" Pinkerton, 1956. Markov chains can be nested or constrained for more complex interactions Collins and Laney, 2017. In Hiller and Isaacson's Hiller and Isaacson, 1959 fourth movement of the Illiac suite, they use Markov chains with a table of possible intervals stretching from unison to octave. In the latter sections of the movement, they introduce additional restrictions to add memory to the method through higher-order Markov chains that reference previously generated music. Transition matrices can be built from very little data, such as short improvisations Pachet, 2003, but training over a whole corpus is also viable. Other systems configure Markov chains to take additional inputs into account, such as Allan, Allan, 2002 who generates harmonies to given melodies in the style of Bach. This makes Markov model-based systems very flexible and relatively lightweight.

	O	C	D	E	F	G	A	B
O	0.38	0.17	0.10	0.10	0.06	0.13	0.03	0.02
C	0.36	0.23	0.13	0.07	0.02	0.10	0.03	0.07
D	0.26	0.20	0.21	0.19	0.03	0.06	0.01	0.05
E	0.22	0.15	0.18	0.16	0.16	0.12	0.01	0.00
F	0.15	0.00	0.14	0.35	0.14	0.20	0.01	0.01
G	0.29	0.14	0.00	0.16	0.06	0.26	0.08	0.00
A	0.17	0.05	0.07	0.00	0.02	0.36	0.15	0.17
B	0.18	0.30	0.12	0.01	0.01	0.08	0.21	0.08

Figure 2.2: Transition matrix from Pinkerton's 1956 "Banal Music Generator". Probability of a pitch (row) following on a pitch (column), likely pairs are marked in yellow. The probabilities are calculated from a set of 39 nursery rhymes.

2.4.4 Other statistical approaches

Music generation has also been attempted with other means, such as metaheuristic search for harmony or melody Altay and Alatas, 2018. In Morpheus Herremans and Chew, 2019 the developers use variable neighbourhood search (VNS) to generate polyphonic pieces following a tension profile for longterm structure. Closely related to this approach are evolutionary and genetic algorithms such as Politio et al. Polito et al., 1997 model of 16th century counterpoint as a multi-population problem. Here three separate populations of agents generate instructions focusing on different musical aspects such as harmony or imitation and are evaluated based on individual performance and on symbiotic performance together with the other agents over multiple generations.

2.4.5 Early Neural Net based systems

Music generators based on neural nets were introduced as early as 1989. Todd et al. Todd, 1989 generate melodies using a fixed window for a conventional feed-forward neural network but also introducing a feedback loop feeding the network's previous state to the next iteration. This forshadows future work based on RNN's such as CONCERT Mozer, 1994 a 1994 RNN trained to generate melodies based on datasets of Bach chorales, waltzes and european folk songs. There are also hybrid systems such as HARMONET Hild et al., 1991, an RNN based music generator for re-harmonizing Bach chorales. It merges the RNN with a symbolic rule-checking algorithm. More recent RNN based music generators such as FolkRNN Sturm and Ben-Tal, 2016, a melody generator trained on irish folk songs, use long short term memory (LSTM')s or gated recurrent units (GRU's). Generative Advesarial Networks (GANs) and Variational

Autoencoders (VAEs) were and still are popular architectures for music generation. Civit et al., 2022

2.5 Deep learning for music generation

State-of-the-art music generation, including many commercial applications, leverages the advances in language and image generation achieved in the last five years. Two distinct approaches, namely autoregressive, transformer-based and diffusion-based approaches dominate.

2.5.1 Transformers - sequence modelling without recurrence

Autoregressive music generation is inspired by successful natural language modeling tasks powered by the transformer model. Transformers were originally developed for the task of language translation.Vaswani et al., 2017 They keep the self-attention mechanism which was deployed prior in LSTMs for seq2seq tasks Sutskever et al., 2014 but replace the recurrent connection with positional embeddings and masked attention. This allows the model to train on all tokens in parallel as opposed to one token at a time. At each layer a token is contextialized with the other tokens, via a paralell multi-head attention mechanism the signal of relevant tokens is enhanced, while the signal of other tokens is diminished. The transformer comes in several different configurations. The original transformer contains both encoder and decoder layers - see figure **??**. Often tasks relating to sequence understanding such as music classification use a encoder only architecture. The BERT series of language models Devlin et al., 2019 and music understanding models such as MusicBERTZeng et al., 2021 are examples. On the other hand sequence generation tasks often employ a decoder only architecturs, this includes the GPT-series Radford et al., 2019 and many music generators such as MusicGenCopet et al., 2023. The transformer architecture is the baseline for all large language models currently in use but has also been applied to modalities beyond text including images (they often form the backbone diffusion models).
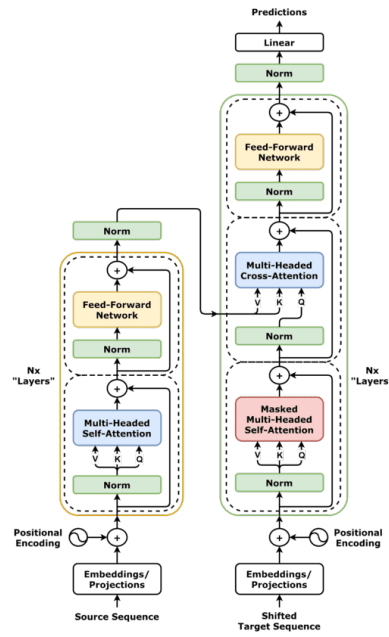
Figure 2.3: Schema of the full transformer encoder-decoder architecture
[a]= By dvgodoy - CC BY 4.0, https://commons.wikimedia.org/w/index.php?curid=151216016

2.5.2 Diffusion models - spectrograms and piano-rolls

Diffusion models are widely used for image and audio generation. Diffusion models learn to remove noise from a distribution (i.e an image). Random noise is added to an image, and the model learns to undo this addition. During inference the model starts with random noise (often accompanied by a guiding text prompt) and undoes it until it arrives at a clear image. AudioLDM Liu et al., 2023, and StableAudio Evans et al., 2024 are diffusion models that operate based on continuous audio encoding by a variational autoencoder. Diffusion models have also been used to generate symbolic music. Polyffusion Min et al., 2023 uses image representations of piano rolls and adapt their diffusion model for various tasks, inpainting, accompaniment and melody generation and generation based on a given chord sequence or texture.

In this thesis we will be targeting transformer models. Transformer models are tremendously popular and behind many of the state of the art results in music generation. There is a tremendous amount of interest in fine-tuning the models, and many methods developed to do so efficiently, including methods to add control. There is a fairly large selection of open-source models that generate music and very good infrastructure to support training and evaluation of those models.

2.6 Representation of music

2.6.1 Symbolic vs Audio

Music can be digitally represented in two ways, either directly as an audio rendition or symbolically as a set of instructions. Working with different representations comes with different drawbacks and benefits for music generation. There are various types of symbolic representation, but the most common consist of discrete sequences of musical elements such as pitch or duraition. Working with audio theoretically gives access to all audible qualities of music, including detailed informatino on instrumental timbre or acoustic setting. In symbolic music this is typically restricted to pitch, duration and instrumentation, which can be extended to include formatting information such as bar-lines etc. Symbolic data is also far less available than audio data, many symbolic music datasets are created by compiling hand-transcribed music. High-quality automatic transcription remains an unsolved issue.Ji et al., 2024Chen et al., 2024 However, symbolic music gives more direct access to many higher-level musical features such as chord progressions, melodies, and instrumentation. When working with audio these features first have to be extracted, which often requires working with additional deep learning models. Another consideration to take into account is size. Raw audio is significantly larger than a corresponding digital score. In addition, rendered audio is difficult to edit once generated.

2.6.2 Tokenisation

Sequences are typically transformed into tokens, a numerical representation of data, to be handled by a machine learning algorithm. Audio-based music generation uses tokenization also as a way to condense audio while retaining its semantic content. Jukebox Dhariwal et al., 2020 uses a variational autoencoderKingma and Welling, 2014 with a discretizing bottleneck (VQ-VAE) to create tokens from audio. Musicgen Copet et al., 2023 tokenizes audio using the previously developed Encodec model for audio compression, which similarly to VQ-VAE learns a highly condensed discrete representation of audio Défossez et al., 2023. These condensed encodings are crucial for generative modeling.

2.6.3 Symbolic Tokenisation

Unlike audio, symbolic music is typically already highly condensed, and a variety of different tokenizers exist for different tasks. Fradet et al., 2021. Common ways of tokenizing symbolic music align closely with the msical instrument digital interface (MIDI) standard, with individual tokens encoding different midi-events such as note-on, note-off, velocity. The REMI+ Y.-S. Huang and Yang, 2020 tokenisation expands on MIDI-based tokenisation with tokens for bar

and position, which is designed to help capture recurring musical patterns. The PerTok tokenizer designed by Lemonaid[2] encodes micro timings and offsets designed to capture the full spectrum of rhythm in musical performances. These extensions come at a cost, the resulting sequences can become very long, which adversly affects the modelJi et al., 2024. Attempts have been made to condense multiple tokens, such as compound word, or nested tokens.Ryu et al., 2024. In MMT Dong et al., 2023, tuples of midi-data are combined into single tokens. Our tokenisation varies by task, but in general encodes events into single characters. This is then combined with a byte pair tokenizer, which combines the most common byte-pairs (extracted in a prior run over the input data) into compound tokens until the vocabulary size is exhausted. The vocabulary size indicates the size of the embedding layer, it is orders of magnitude larger than the characters used for the single events.Sennrich et al., 2016. This approach mimics the compound word approach by condensing the most common combinations into single tokens. This drastically shortens the sequence length by about a third and improves modeling. More in the appendix **??**

See table **??** in the appendix for a more elaborate display of representations used in symbolic music generators

Herremans and Chew, 2019

2.7 Control

Control is an essential aspect of any generative model. Without control, even the best generative models producing beautiful music would be of very limited real-world use. Control allows generative AI tools to become proper collaborative systems, and generate for a wide array of scenarios. In music generation control covers essentially all musical parameters. Parameters vary by representation, symbolic music for instance leaves very little room for any type of timbre control (aside from instrument selection). "Raw" audio models such as Stable Audio (Evans et al., 2024) can for instance be controlled for acoustic settings (i.e jazz music playing in a busy restaurant, in a *large cathedral*, or *through an intercom*), something that is simply not represented in symbolic music. For musical parameters represented in symbolic music, there are different approaches to classifying them. We can differentiate between global and local features Van Kranenburg et al., 2013 or deep vs surface-level features Blacking, 1971.

Global features in music generation encompass high-level descriptors such as genre, function, and instrumentation/orchestration. Global features also incorporate summarizing features, that are calculated from the music itself. As an example, McKay's McKay, 2004 general-

---

[2]https://www.lemonaide.ai/

purpose symbolic music classification system defines over 104 global features, 37 of them used for melody descriptors such as probability distributions of note durations, intervals, and pitch classes. These global features can then be used to infer other high-level descriptors such as genre if it's unknown, they can also support music retrieval systems Van Kranenburg et al., 2013. **Local features** describe individual sequences such as melodic, rhythmic, or harmonic sections. They are more information-rich, which may explain their superior performance in music retrieval Van Kranenburg et al., 2013.

Also relevant in the music generation context are abstract vs concrete features. Music generation can take abstract concepts such as genre Rouard et al., 2024, Lu et al., 2023 or emotion Tan and Herremans, 2020 Lu et al., 2023 into account, which in turn effects more complex dynamics of concrete features. To illustrate: In Music FaderNets Tan and Herremans, 2020 the authors use a variational autoencoder to disentangle the abstract concept of arousal, into several concrete features including rhythmic density, note-density, tempo and dynamic, key. This allows them to change the abstract variable arousal, which cascades into a change of underlying features. This type of disentanglement can aid in situations where labled musical data with the abstract feature is less available, it can also aid in creating more accessible and interpretable generative models.

In the context of this thesis we differentiate between global features, and fine-grained or time-varying features Rütte et al., 2023. What is time-varying or global is highly context dependent, a piece may have one time-signature and tempo as is assumed in Lu et al., 2023 or it may vary over time as is suggested in Rütte et al., 2023 or Y.-S. Huang and Yang, 2020. Other time varying controls could be chords Rütte et al., 2023Wu et al., 2023Lan et al., 2024Min et al., 2023, melody Copet et al., 2023Min et al., 2023 or texture Min et al., 2023. For the target application in an MACT - game, time-varying controls are necessary to provide a change in music that triggers a change in the patients improvisation. For a more complete list of the types of controls in many of the common symbolic music generators refer to table **??** in the appendix.

2.7.1 Rhythmic control

The types of control exercised over rhythmic components varies by representation as discussed in **??**. In CocoMulla Lin et al., 2024 generated audio is conditioned with drum tracks and a piano-roll. Similarly in JASCOTal et al., 2024 drums are used for conditioning. In MusicConGen.Lan et al., 2024control for rhythm is added through tracking beats and downbeat. MusicControlNetWu et al., 2023 adds beat and downbeat conditioning to an audio diffusion model. For symbolic systems control of tempo and meter is relatively common Rütte et al., 2023, Y.-S. Huang and Yang, 2020, Lu et al., 2023. More fine grained control over rhythm is sometimes deployed through note-density (both vertical and horizontal)Rütte et al., 2023,Y.

Huang et al., 2024. Another option is control over texture, which merges harmonic and rhythmic elements. In PolyffusionMin et al., 2023, texture is encoded by a pretrained variational auto-encoderWang, Wang, et al., 2020. Another approach Zhu et al., 2024 involves passing the piano-roll as factor to guide the diffusion process.

2.7.2 Target Features

In order to extend LastMinuteGig with generated music containing appropriate cues we are targeting control of rhythmic patterns. The most simple way of targeting rhythmic patterns is through note density, implemented as a bar-level instruction of how many notes the following bar should contain. A second simple target feature is note variability (unique notes per bar over number of notes per bar). Both note-density and note-variability are simple time-varying features that are easily extracted, tokenized and introduced to the algorithm.

A potential approach, novel to music generation, would be control mechanisms using either spectral weight or inner metric weight. Here the weight profile is passed as a control mechanism either globally or on a per bar or per section level. *Comment: Introducing the control at a per bar level would be a bit odd since the idea of metric weight is about extracting weights not reflected in the symbolic grid, but it is a sensible way to introduce it to the model in bite-sized chunks able to reflect local change.* Inner metric and spectral weight weight has been successfully applied to dance music classification Chew et al., 2005, meter detection Haas and Volk, 2016 and music retrieval Volk et al., 2009. This suggests that it is a powerful feature for explaining various rhythmic aspects, which may make it a good guiding mechanism.

2.8 Implementation of control

Adding control to generative models can be split into three approaches. 1: Choice of architecture and training data, 2: fine-tuning approaches, and 3: external conditioning or guidance.

2.8.1 Control through architecture

The choice of architecture lends itself to different types of control. Transformers are next token predictors, that predict based on the prior sequence. The default training paradigm allows for conditioning with a user defined musical (or audio) sequence. This is true for both audio based models such as MusicGen Copet et al., 2023, Jukebox Dhariwal et al., 2020 and MusicLM Agostinelli et al., 2023 as well as symbolic music models such as MMT Dong et al., 2023, MusicTransformer C.-Z. A. Huang et al., 2018 and MusicBERT Zeng et al., 2021.

Diffusion models are quite flexible compared to transformers, the same model can be used for inpainting, continuation and depending on the representation melody and accompaniment

generation through masking.Min et al., 2023Rombach et al., 2022

2.8.2 Control through training conditioning

Control can be added through training data. MusicGenCopet et al., 2023, a recent text-to-music (audio) transformer is trained on 20000 hours of licensed music from shutterstock and pond5 [3] which includes textual descriptions and tags for genre, tempo, and other factors such as instrumentation. Control is achieved through the joint training of a text description and music. An example description is provided below:

Inspirational dramatic background music! Perfect for trailer, background, advertising, historical film, movie about superheroes, teaser and many other projects! [4]

Text-based control, while user-friendly and accessible to non-musicians, is inherently vague. Levels of detail and choice of words vary widely by dataset, even with standardized tags such as genre and tempo. Specialized datasets such as MusicCaps Agostinelli et al., 2023, which contains 5500 text music pairs, with 10 seconds of music alongside a free text description and a list of aspect tags created by human experts still suffer substantially from subjectivity Lee et al., 2023.

For this reason, the creators of MusicGen Copet et al., 2023 add melody conditioning alongside text conditioning and train their model jointly with the chromagram of the melody alongside the text.

In MusicGenStyle Rouard et al., 2024 perform classifier-free guidance to add style conditioning to MusicGen. They train a music-style encoder that transforms a random subsample of a given reference audio track into tokens that are combined with the embeddings of the text-description. Both the style tokens and text tokens are provided as prefix to the model. The conditioner and the MusicGen transformer are trained jointly on the entire dataset.

The creators of FIGARORütte et al., 2023 enable fine grained control over instrumentation, note density, average pitch and volume on a bar-by-bar basis, in a symbolic music generator through joint conditioning while training.

2.8.3 Adding control through fine-tuning

Both the melody conditioning of MusicGen Copet et al., 2023 and the style conditioning of MusicGenStyle Rouard et al., 2024 retrain the entire MusicGen model on the entire dataset which comes at considerable cost. Fine-tuning or transfer learning is another method through

---

[3]https://www.shutterstock.com/music and https://www.pond5.com/
[4]https://www.pond5.com/royalty-free-music/item/95908062-inspiring-dramatic-epic-background-cinematic-music

which models can be trained but at considerably smaller cost and using less data. This is particularly useful and widely used in the language domain to adjust large language models for niche use-cases, where the available data may simply not be sufficient to train a large model from scratch. In the examples of MusicGen and MusicGenStyle the availability of data was not a limiting factor since the controlling elements, melody and style can be inferred from the training data. However fine-tuning is also helpful for adding new control mechanisms.

MusiConGen Lan et al., 2024 is a fine-tuned variation of MusicGen which adds rhythm and chord control. They propose the jump-finetuning mechanism, where the original model with 1.5 Billion parameters and 48 self-attention layers, is split model into blocks consisting of 4 self-attention layers. They refine the first layer of each block, freezing the remaining layers. Additionally, they apply adaptive in-attention to the first 9 blocks, where the output of the transformer is augmented with copies of the original condition. As a result, only a quarter of the original parameters are tunable, which enables training on consumer GPUs on just 250 hours of music sourced from YouTube (as opposed to 20000 hours). In Coco-Mula Lin et al., 2024 the authors adjust a LLAMA adapter with just 4% of parameters, keeping all original MusicGen parameters frozen, and training only the adapter on a small dataset of 300 songs to add drum and chord conditioning.

MuseBarControl Shu et al., 2024 is a fine-tuned version of MuseCocoLu et al., 2023 which extends the global controls with fine-grained bar level control for music-generation. They compare several approaches. In the first they augment the prompt (which is generated from text) with additional tokens for bar-wise control of chords, and adjust the loss function to incoperate that. In the second approach they introduce two novel methods, first, they pre-adapt the new parameters (introduced by the lora adapter) to a separate classification task, an auxiliary task . The model classifies whether the a section of music corresponds with the control tokens, the body of the model is trained together with a classification head (which is removed after auxiliary task training. In the third step they introduce counterfactual loss where the difference in negative log likelyhood conditioned on the original and changed attribute is maximized, which reinforces the models attention to the control. They find that the combination of the three strategies, pre-adaptation on a separate task followed with counterfactual-loss and prompt augmentation yields the strongest model.

2.8.4 Adding control through guidance

There are also other methods that do not involve any finetuning or retraining of the original model. Adding control through rule labels, or control tokens as in Rütte et al., 2023Lan et al., 2024 does require some amount retraining, which is not always feasible, and adding many different types of control may deterioate the model. In these cases, guidance can be used to

steer the model towards a certain output. In SMITIN Koo et al., 2024 the authors use inference time intervention to guide a model towards a certain output with respect to certain goals. The goals explored are the presence of certain instruments (piano/drums/bass/guitar) in the mix and the quality/realism of the music. The authors train linear probes that learn to associate a state of the attention heads with the stated goal. Then the attention heads are steered in the direction of the probe's output, which increases the probability of the desired quality being present in the generated music.

In Diffusion models, the output is sampled over several steps, at each of these steps it is possible to intervene with guidance to direct the sampling towards a certain goal. In Y. Huang et al., 2024, each sampling step is repeated several times, and each time the sample that follows a set of rules most closely is chosen. ControlNet Zhang et al., 2023 adds spacial control to image generators allowing the guidance of image generation using sketches, poses, edges and depth maps without retraining. MusicControlNet Agres et al., 2021 adapts this approach to music generation adding control for time varying factors, melody, dynamics and rhythm.

2.9   Evaluation

How to evaluate generated music is still an open research question. There are no standardized methods according to which evaluations happenYin et al., 2023. In the context of music generation there are several proposed frameworks to evaluate music. Typically we differentiate between subjective and objective evaluations.

For subjective approaches the methods vary widely Xiong et al., 2023. There are simple Turing-type evaluations that test how distinguishable generated and human written music are. Then there are subjective query metrics, where typically likert ratings of different parameters are collected.Min et al., 2023 There are tournament style surveys, where the number of winning pieces are tallied for each approach.C.-Z. A. Huang et al., 2018Rütte et al., 2023 Finally there are expert evaluations (which can also include likert ratings) but also analysis of the produced score and musical structure. Sturm and Ben-Tal, 2016 These evaluations are often paired with statistical hypothesis testing. Rütte et al., 2023

Automatic evaluation of generated music include model specific metrics and different musical metrics Xiong et al., 2023. Model specific metrics are generic evaluations of a models success to approximate training data, these will vary depending on the model and are not indicative of stylistic success. Examples of this are Negative Log Likelyhood C.-Z. A. Huang et al., 2018, Root Mean Square Error Rütte et al., 2023 or PerplexityRütte et al., 2023. Musical metrics typically involve comparing a set of generated music to a set of real music, there are plenty

of musical similarity measure techniquesGurjar and Moon, 2018 for a large variety of different use-cases i.e music retrieval, cover, genre and artist detection. A popular comparative metric is calculating the Kulback Leibler (KL) divergence between two datasets with respect to certain metrics i.e count of intervals or unique pitch-classes. However to obtain the divergence one has to select specific features that may only capture a subset of the desired properties. Similar issues arise with other distance metrics i.e cosine similarity, earth movers distance or maximum overlapping area.

Especially in the audio domain, additional AI models are often used for evaluation. Music-Gen Copet et al., 2023 uses additional classifiers to generate labels for the music and calculates the KL-divergence between the generated labels. Additionally they calculate the Frachet Audio distance, a measure devised to calculate the plausibility of audio (for music enhancement purposes) compared to a large set of studio recordingsKilgour et al., 2019. Finally they use the CLAP-score which compares the corresponding text description to the latent representation of the generated audio, with text-description of the generated audio with the reference audio. Elizalde et al., 2023

For this thesis we are interested in two factors, first the plausibility of the generated music, and second the success of the control. How the success of control is evaluated depends on what is controlled for, Examples of controlled parameters and how they are evaluated are as follows:

Note Density. (how many notes per bar). Root mean square error (RME) between generated vs target notes per bar. This is the approach to compare note density used in Rütte et al., 2023

Note Variability (number of unique pitch classes, normalized by number onsets) - RME between generated and target.

Rhythmic patterns. Partial Similarity Volk et al., 2009 between target and generated music:

Evaluating the plausibility of generated music is more complex, and there is no one method that has been proven superior. Possibly the plausibility of the music will be evaluated with a (small) subjective study. In this scenariou we would follow Dong et al., 2023, Yu et al., 2022 and Chen et al., 2024 and collect likert ratings on questions targeting Coherence Richness Arrangement and Consistency . Theese are would be paired with questions on musical background and preferences of the participant. For objective rating of the plausibility of the music we will follow the approach by Min et al., 2023 where the KL-divergence between the corpus of generated and a corpus of original music is calculated, most likely KL divergence over a set of extracted features, such as a pitch classes and chords. For a more complete list of different evaluation methods used on symbolic music generators, refer to the notes in the abstract **??**.

3. Research Plan

3.1 Research Gaps and Opportunities

There are several different methods to add control to music generation models, through training data, architecture choice and through various methods of fine-tuning as discussed in chapter **??** However there are few comparisons in literature between the different methods of adding control, aside from pointing out the general advantages of not retraining the entire model Koo et al., 2024Lan et al., 2024Lin et al., 2024. Adding control to trained symbolic models is less common in literature as opposed to audio generators. Symbolic music generators tend to be smaller, and are often developed in non-commercial settings, training one from scratch is a often smaller effort. However adding control to a symbolic model presents a unique oppertunity to introduce complex symbolic control mechanisms, while maintaining the flexibility of symbolic music, which can be integrated into most music production environments.

Specifically this thesis investigates adding a certain type of control of rhythm. While generative models feature control of drums Lan et al., 2024, tempo, note-density or meter Rütte et al., 2023, Y.-S. Huang and Yang, 2020, control using inner metric weight, or any complex symbolic characterization of rhythm is novel. However given the success of this characterisation in classification and music-identification tasks it is a promising control feature. The closest approximation is that of texture Min et al., 2023 which has been deployed to some success in diffusion model using VAE disentanglement of harmony and melody. Finally the topic of MACT games offers plenty of opportunity for exploration. Extending the LastMinuteGigChalkiadakis, 2022 application to include generated music, and evaluating the effects of that on enjoyment, engagement and effect is a third area of opportunity. To summarize:

1. Research Opportunity 1: Comparison between different methods of adding control.

2. Research Opportunity 2: Adding Control for rhythmic patterns to a trained symbolic music model

3. Research Opportunity 3: Integration of generated music in MACT application.

Below I outline both the target models and each individual step. The timeline is outlined in the last section, and is quite ambitious. All of the three parts can be decoupled, i.e if the first part is not successful, the second and third parts can still be completed.

3.2   MusicLang Model Suite

For these experiments I am collaborating with the developers of musiclang[1] a suite of open source symbolic music generators who provide the base models. I choose MusicLang for a variety of reasons.

3.2.1   Music Lang Core

Musiclang's core model is a transformer based on LLAMA 2 trained on the Lakh Midi DatasetRaffel, 2016. Its part of active open source community, well documented and integrated into up to date libraries. If the thesis is successful, I would love to contribute the results here, where they are visible and be of use to others. MusicLang generates high-quality symbolic music, while being relatively small in terms of trainable parameters, fit to run on a consumer machine. The model is already integrated into a custom composition environment, and the developers are activley working to integrate it into a VST plugin to be hosted in the vast majority of music production environments. Finally it already contains controls for chord progression, instrumentation, range, and harmonic coherence, and can generate interpolations, additional instrumental lines and continuations of pre-composed music. This is a useful fallback in case the control of rhythm is not sufficient. There are other control dimensions that can introduce stimulus to the MACT game. For a more complete evaluation of alternative trained symbolic music models see the appendix **??**.

3.2.2   BassCraft a small transformer model

For the initial experiments comparing different methods of adding control (Opportunity 1) we are using a very small transformer model: BassCraft a GPT2 Radford et al., 2019 based transformer. It has an embedding size of 256, 4 attention heads, 4 hidden transformer layers, and a total of 7 million trainable parameters. (For contrast, Llama 3, with the most recent version released in December 2024, comes at different sizes ranging from 1 to 405 billion trainable parameters. While a model like this makes it easier to create a proof of concept due to smaller training costs, there also additional real-world advantages. The model is small and flexible enough to be integrated into a composition environment.

Basscraft is trained to generate a bassline to an existing excerpt of music. It is trained using the lakh-midi dataset Raffel, 2016. For training, songs with bass-lines are selected (based on the presence of particular MIDI-instrument channels). The tracks are partitioned into snippets between 1 and 16 bars long. The bass-lines are separated from the remaining track and matched as potential output.
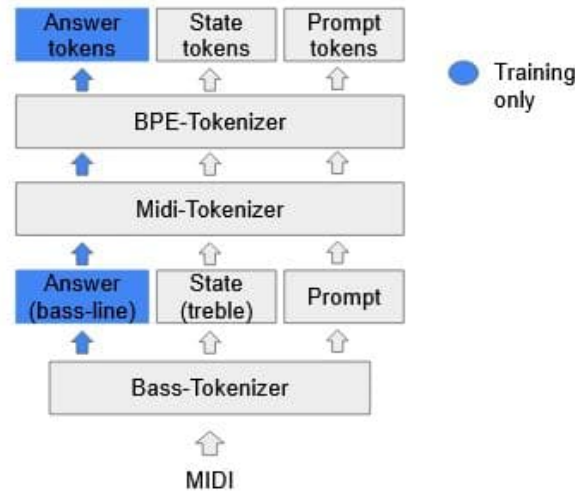
---

[1]https://musiclang.github.io/

Figure 3.1: Preprocessing/tokenization of the original basscraft model

3.3 Comparing methods of adding control ca 1.5 months

3.3.1 Approach 1 - Vocabulary Extension

Vocabulary expansion is the process to add new vocabulary to a transformer model. The method depends on several factors including which tokenizer is used, the set of new words and how it differs from the prior set. I.e is it a new language with no overlap, or a domain specific subset of a language where you may want to add common domain specific words.

In vocabulary expansion one maintains the prior tokenizer, to keep pretrained knowledge. The tokenizer used for the Basscraft model is a modified byte pair encoding (BPE) tokenizer Sennrich et al., 2016, which is trained starting from a general alphabet, and greedily merges the most common pairs of tokens until the target vocabulary size is reached (16000). It is difficult to simply add new tokens here, as it would require retraining the BPE tokenizer, which will transform the embedding layer making it unusable. Common approaches are merging tokenizers, resizing the vocabulary to add new tokens while keeping the prior embeddings intact or replacing vocabulary items. We choose vocabulary replacement of unused tokens with the new control tokens. This will inevitibally extend the sequence size, since the new tokens are not merged in the BPE process, which may adversly effect the model performance. However this will allow us to maintain the same embedding size, while still benefiting from the model's prior training.
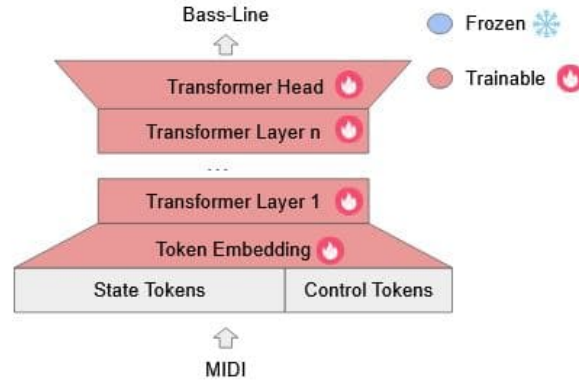
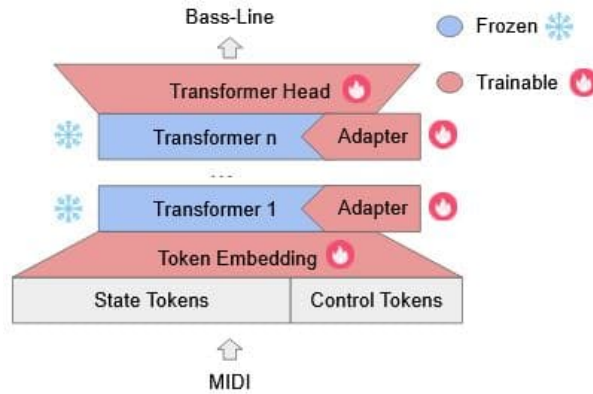**Figure 3.2:** Vocabulary transfer with full fine tuning

Figure 3.3: Vocabulary transfer with parameter efficient fine tuning

3.3.2 Approach 2 - Integrating of control tokens

In this experiment we adjust the approach used in Coco-Mulla Lin et al., 2024 inserting the embedded condition prefix into the last frozen layers of the model. The embedding is learnable. In contrast to the previous method we are not working with tokens as input tokens, we are incorporating them into the model after c layers. This method has the benifit that we don't need to worry about tokenisation and the vocabulary of the original model, as those remain unchanged. The difficulty is learning the embedding of the control tokens.
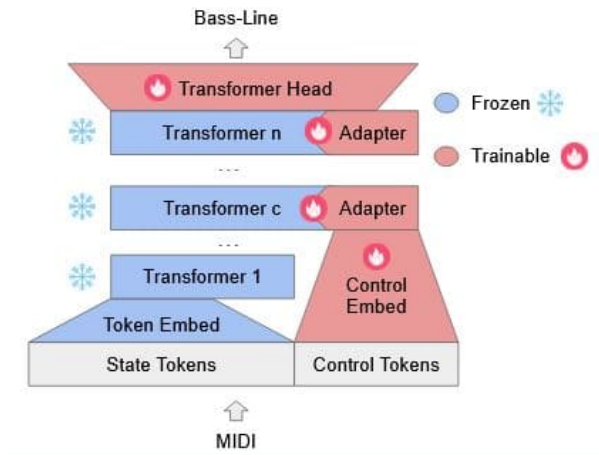
Figure 3.4: Integration of control tokens with parameter efficient fine tuning

3.3.3 Approach 3 - Post Hoc Conditioning

This is the approach used by SMITINKoo et al., 2024. ->

One potential problem: This may be very difficult to implement. SMITIN only has binary scenarios (presence of certain instruments), our current target variables are larger. What is nice about this approach is that it is very flexible, can be used in conjunction with other methods, and roughly emulates the guidance mechanisms used in diffusion models.
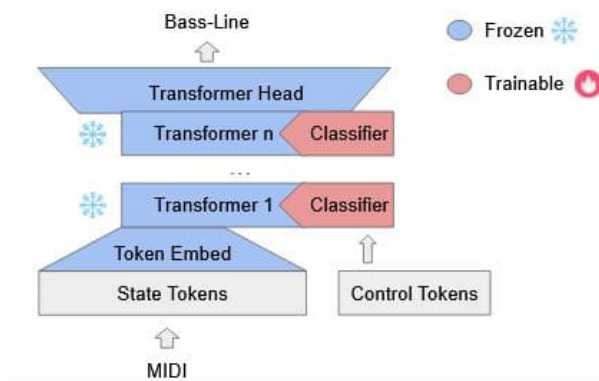

Figure 3.5: Integration of control using inference time interference

If these experiments are not successful, we can follow the approach of Shu et al., 2024 and try additional training using auxiliary tasks and using a modified (counterfactual) loss function.

Other potential avenues to explore in this section (if time remains) is how controls stack. Adapters for the different target controls can be trained separately, then merged.

3.4   Application to larger model 1.5 months

The first step in this experiment is extracting and tokenizing either spectral or inner metric weight which is then tokenized and passed to the MusicLang Llama 2 model. They will be integrated with the most successful method determined above.

3.5   Integration into LastMinuteGig 2 months

This project is motivated by extending the music engine of LastMinuteGig. Chalkiadakis, 2022. The application currently employs a very simple algorithm. I will outline it below.

- 1: Randomly choose key, rhythmic pattern, chord progression and tempo from a pool of possible values.

- 2: Play corresponding percussion audio clip (depends on tempo and rhythmic pattern)

- 3: When user plays the button trigger the guitar sound of the current chord.

- 4: After 8 bars - make a random change (to rhythm, tempo, pause)

This music engine will be changed in the following way. Most of the generative process will be done asynchronously.

- 1: Create $n = 100$ musical pieces with different control settings

- 2: For each piece

- 3: Extract chords from prompt (assuming high control success) - save mapping $m1$

- 4: Extract change of rhythmic pattern (or tempo or meter or instrumentation since these are already controllable in the musiclang model) - save mapping

- 5: Render symbolic music to audio

During game play

- 1: Load random audio clip and mappings.

- 2: Play associated guitar chords when player taps.

- 3: Register changes in tapping on musical changes.

- 4: Repeat when audio clip has finished playing.

This will be followed by a user study on healthy participants following the original proto-colChalkiadakis, 2022
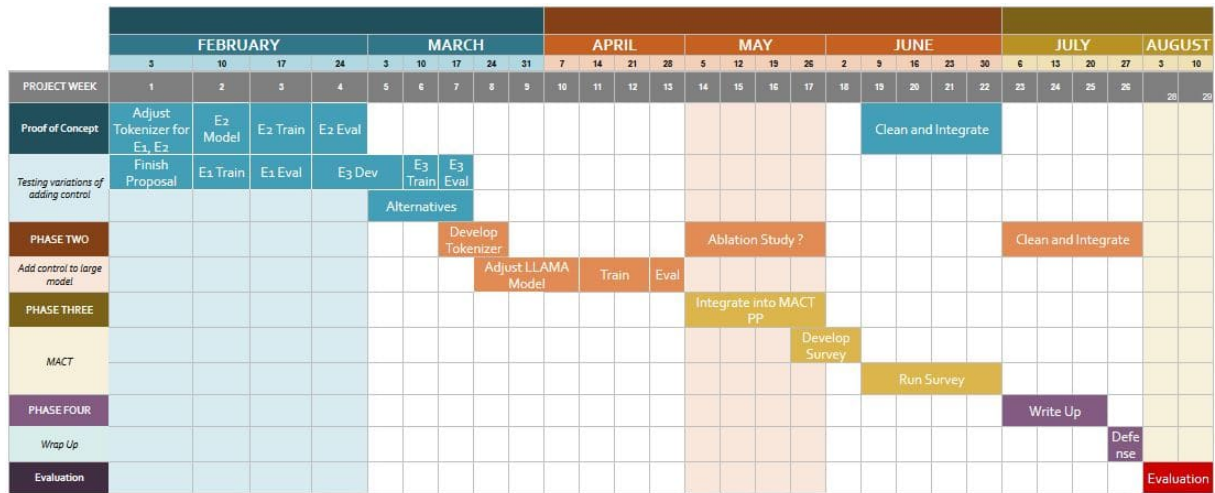
3.6   Thesis Timeline

Figure 3.6: Thesis project plan

Bibliography

Rogers, K., & Weber, M. (2019). Audio habits and motivations in video game players. *Proceedings of the 14th International Audio Mostly Conference: A Journey in Sound*, 45–52. https://doi.org/10.1145/3356590.3356599

Chalkiadakis, E. (2022, January). *Developing and evaluating a musical attention control training computer game application.* [Master's thesis, Utrecht University]. https://studenttheses.uu.nl/bitstream/handle/20.500.12932/500/Master%20Thesis%20-%20Developing%20and%20evaluating%20a%20Musical%20Attention%20Control%20Training%20computer%20game%20application%20%285892252%29.pdf?sequence=1%5C&isAllowed=y

Copet, J., Kreuk, F., Gat, I., Remez, T., Kant, D., Synnaeve, G., Adi, Y., & Défossez, A. (2023). Simple and controllable music generation. *Thirty-seventh Conference on Neural Information Processing Systems*. https://doi.org/10.48550/arXiv.2306.05284

Agostinelli, A., Denk, T. I., Borsos, Z., Engel, J., Verzetti, M., Caillon, A., Huang, Q., Jansen, A., Roberts, A., Tagliasacchi, M., Sharifi, M., Zeghidour, N., & Frank, C. (2023). Musiclm: Generating music from text. *arXiv*, (arXiv:2301.11325). https://doi.org/10.48550/arXiv.2301.11325

Ferdinand Meyen, B. R. (2024). "verknallt in einen talahon": So reagiert der talahon-produzent auf die rassismus-vorwürfe. *Bayern 2 Zuendfunk*. https://www.br.de/radio/bayern2/sendungen/zuendfunk/verknallt-in-einen-talahon-produzent-rassismus-vorwuerfe-charts-udio-ki-100.html

Stassen, M. (2024). Suno, with a $500m valuation, has admitted training its ai on copyrighted music. *Music Business Worldwide*. https://www.musicbusinessworldwide.com/suno-with-a-500m-valuation-has-admitted-training-its-ai-on-copyrighted-music-it-just-named-timbaland-as-a-strategic-advisor1/

Nierhaus, G. (2009). *Algorithmic composition (pp 3638)*. Springer. https://doi.org/10.1007/978-3-211-75540-2

Cope, D. (1989). *New directions in music*. Dubuque, Iowa : W.C. Brown. http://archive.org/details/unset0000unse_x4d7

(n.d.). https://suno.com/about

Hadjeres, G., Pachet, F., & Nielsen, F. (2017). Deepbach: A steerable model for bach chorales generation. *Proceedings of the 34th International Conference on Machine Learning (ICML)*. https://doi.org/10.48550/arXiv.1612.01010

Malandro, M. E. (2023). Composer's assistant: An interactive transformer for multi-track midi infilling [arXiv:2301.12525 [cs]]. *2024 International Society for Music Information Retrieval*. https://doi.org/10.48550/arXiv.2301.12525

Pachet, F. (2003). The continuator: Musical interaction with style. *Journal of New Music Research*, *32*(3), 333–341. https://doi.org/10.1076/jnmr.32.3.333.16861

Kite-Powell, J. (2023, February). See how this ai improvised with a musican to create a musical dialogue in real-time. https://www.forbes.com/sites/jenniferhicks/2023/02/25/see-how-this-ai-improvised-with-a-musican-to-create-a-musical-dialogue-in-real-time/

Plut, C., & Pasquier, P. (2020). Generative music in video games: State of the art, challenges, and prospects. *Entertainment Computing*, *33*, 100337. https://doi.org/10.1016/j.entcom.2019.100337

Worrall, K., & Collins, T. (2024). Considerations and concerns of professional game composers regarding artificially intelligent music technology. *IEEE Transactions on Games*, *16*, 586–597. https://doi.org/10.1109/TG.2023.3319085

Bibliography

Djaouti, D., Alvarez, J., & Jessel, J.-P. (2011). Classifying serious games: The g/p/s model. In P. Felicia (Ed.), *Handbook of research on improving learning and motivation through educational games: Multidisciplinary approaches* (pp. 118–136). IGI Global. https://doi.org/10.4018/978-1-60960-495-0.ch006

Agres, K. R., Schaefer, R. S., Volk, A., van Hooren, S., Holzapfel, A., Dalla Bella, S., Müller, M., de Witte, M., Herremans, D., Ramirez Melendez, R., Neerincx, M., Ruiz, S., Meredith, D., Dimitriadis, T., & Magee, W. L. (2021). Music, computing, and health: A roadmap for the current and future roles of music technology for health care and well-being. *Music & Science*, *4*, 2059204321997709. https://doi.org/10.1177/2059204321997709

Park, J.-K., & Kim, S. J. (2021). Dual-task-based drum playing with rhythmic cueing on motor and attention control in patients with parkinson's disease: A preliminary randomized study. *International Journal of Environmental Research and Public Health*, *18*(19), 10095. https://doi.org/10.3390/ijerph181910095

Martin-Moratinos, M., Bella-Fernández, M., & Blasco-Fontecilla, H. (2023). Effects of music on attention-deficit/hyperactivity disorder (adhd) and potential application in serious video games: Systematic review. *Journal of Medical Internet Research*, *25*, e37742. https://doi.org/10.2196/37742

Pasiali, V., LaGasse, A. B., & Penn, S. L. (2014). The effect of musical attention control training (mact) on attention skills of adolescents with neurodevelopmental delays: A pilot study. *Journal of Music Therapy*, *51*(4), 333–354. https://doi.org/10.1093/jmt/thu030

van Alphen, R., Stams, G. J. J. M., & Hakvoort, L. (2019). Musical attention control training for psychotic psychiatric patients: An experimental pilot study in a forensic psychiatric hospital. *Frontiers in Neuroscience*, *13*, 570. https://doi.org/10.3389/fnins.2019.00570

Schlette, L. (2022). *Increasing engagement in a mact game through feedback and adaptability* [Master's thesis, Utrecht University].

Tencer, D. (2024). New ai-powered 'instant' music-making app udio raises $10m; launches with backing from will.i.am, common, unitedmasters, a16z. *Music Business Worldwide*. https://www.musicbusinessworldwide.com/new-ai-powered-instant-music-making-app-udio-raises-10m-launches-with-backing-from-will-i-am-common-unitedmasters-a16z/

Kaba, M. M., River, M. N., & Perry, A. R. (2024, June). Demand for jury trial. https://storage.courtlistener.com/recap/gov.uscourts.mad.272063/gov.uscourts.mad.272063.1.0.pdf

Duan, M., Suri, A., Mireshghallah, N., Min, S., Shi, W., Zettlemoyer, L., Tsvetkov, Y., Choi, Y., Evans, D., & Hajishirzi, H. (2024). Do membership inference attacks work on large language models? [arXiv:2402.07841 [cs]]. *Conference on Language Modeling (COLM)*. https://doi.org/10.48550/arXiv.2402.07841

Carlini, N., Hayes, J., Nasr, M., Jagielski, M., Sehwag, V., Tramèr, F., Balle, B., Ippolito, D., & Wallace, E. (2023). Extracting training data from diffusion models [arXiv:2301.13188 [cs]]. *Proceedings of the 32nd USENIX Security Symposium*. https://doi.org/10.48550/arXiv.2301.13188

Newton-Rex, E. (2024). Suno is a music ai company aiming to generate $120 billion per year. but is it trained on copyrighted recordings? *Music Business Worldwide*. https://www.musicbusinessworldwide.com/suno-is-a-music-ai-company-aiming-to-generate-120-billion-per-year-newton-rex/

Reed, R. (2024). Does chatgpt violate new york times' copyrights? *Harvard Law School*. https://hls.harvard.edu/today/does-chatgpt-violate-new-york-times-copyrights/

Copyright law of the united states title 17, chapter 1, section 107. (n.d.). https://www.copyright.gov/title17/92chap1.html#107

Singh, M. (2024, September). Indian filmmaker ram gopal varma abandons human musicians for ai-generated music. https://techcrunch.com/2024/09/19/indian-filmmaker-ram-gopal-varma-abandons-human-musicians-for-ai-generated-music/

Stassen, M. (2023). "there are now 120,000 new tracks hitting music streaming services each day". *Music Business Worldwide*. https://www.musicbusinessworldwide.com/there-are-now-120000-new-tracks-hitting-music-streaming-services-each-day/

Marechal, A. (2024). Generative ai: Energy consumption soars. *Polytechnique Insights*. https://www.polytechnique-insights.com/en/columns/energy/generative-ai-energy-consumption-soars/

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., & Amodei, D. (2020). Scaling laws for neural language models [arXiv:2001.08361 [cs]]. *arXiv*, (arXiv:2001.08361). https://doi.org/10.48550/arXiv.2001.08361

Holzapfel, A., Kaila, A.-K., & Jääskeläinen, P. (2024). Green mir? investigating computational cost of recent music-ai research in ismir. *25th Int. Society for Music Information Retrieval Conf.* https://drive.google.com/file/d/1rAepoJk1U2R4g3S_AcxdqWDGzRGd7xPg/view?usp=embed_facebook

France-Presse, A. (2016). First recording of computer-generated music – created by alan turing – restored. *The Guardian*. https://www.theguardian.com/science/2016/sep/26/first-recording-computer-generated-music-created-alan-turing-restored-enigma-code

Hiller, L., & Isaacson, L. M. (M. (1959). *Experimental music; composition with an electronic computer*. New York, McGraw-Hill. http://archive.org/details/experimentalmusi00hill

Ebcioğlu, K. (1994). An expert system for harmonizing chorales in the style of j. s. bach. *Understanding Music with AI: Perspectives on Music Cognition, eds. M. Balaban, K. Ebcioğlu, and O. Laske*, 145–185.

Polito, J., Daida, J. M., & Bersano-Begey, T. F. (1997). Musica ex machina: Composing 16th-century counterpoint with genetic programming and symbiosis (P. J. Angeline, R. G. Reynolds, J. R. McDonnell, & R. Eberhart, Eds.) [Book Title: Evolutionary Programming VI]. *Evolutionary Programming*, *1213*, 113–123. https://doi.org/10.1007/BFb0014805

Todd, P. M. (1989). A connectionist approach to algorithmic composition. *Computer Music Journal*, *13*(4), 27–43. https://doi.org/10.2307/3679551

Mozer, M. C. (1994). Neural network music composition by prediction: Exploring the benefits of psychoacoustic constraints and multi-scale processing. *Connection Science*, *6*(2–3), 247–280. https://doi.org/10.1080/09540099408915726

Chakraborti, T., McCane, B., Mills, S., & Pal, U. (2020). Coconet: A collaborative convolutional network [arXiv:1901.09886 [cs]], (arXiv:1901.09886). https://doi.org/10.48550/arXiv.1901.09886

Xenakis, I. (1992). *Formalized music thought and mathematics in composition*. Pendragon Press. https://monoskop.org/images/7/74/Xenakis_Iannis_Formalized_Music_Thought_and_Mathematics_in_Composition.pdf

Featherstone, E. (2017). Introducing the next generation of music makers. *The Guardian*. https://www.theguardian.com/small-business-network/2017/aug/29/computer-write-music-jukedeck-artificial-intelligence

Yang, L.-C., Chou, S.-Y., & Yang, Y.-H. (2017). Midinet: A convolutional generative adversarial network for symbolic-domain music generation. *18th International Society for Music Information Retrieval Conference (ISMIR)*. https://doi.org/10.48550/arXiv.1703.10847

Sturm, B. L. T., & Ben-Tal, O. (2016). Folk the algorithms: (mis)applying artificial intelligence to folk music. In E. R. Miranda (Ed.). Springer International Publishing. https://doi.org/10.1007/978-3-030-72116-9_16

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*. https://doi.org/10.48550/arXiv.1706.03762

Dhariwal, P., Jun, H., Payne, C., Kim, J. W., Radford, A., & Sutskever, I. (2020). Jukebox: A generative model for music. *arXiv*, (arXiv:2005.00341). https://doi.org/10.48550/arXiv.2005.00341

Christine, P. (2019, April). Musenet. openai.com/blog/musenet

Yin, Z., Reuben, F., Stepney, S., & Collins, T. (2023). Deep learning's shallow gains: A comparative evaluation of algorithms for automatic music generation. *Machine Learning*, *112*(5), 1785–1822. https://doi.org/10.1007/s10994-023-06309-w

Collins, T., & Laney, R. (2017). Computer-generated stylistic compositions with long-term repetitive and phrasal structure. *Journal of Creative Music Systems*, *1*(22). https://doi.org/10.5920/JCMS.2017.02

Huang, C.-Z. A., Vaswani, A., Uszkoreit, J., Shazeer, N., Simon, I., Hawthorne, C., Dai, A. M., Hoffman, M. D., Dinculescu, M., & Eck, D. (2018). Music transformer [arXiv:1809.04281]. *arXiv*, (arXiv:1809.04281). https://doi.org/10.48550/arXiv.1809.04281

Mirelman, S. (2013). Tuning procedures in ancient iraq. *Analytical Approaches to World Music 2*, (2), 43–56.

Fux, J. J. (1725). *Gradus ad parnassum*. Catholicae Majestati Aulae-Typographi. https://www.loc.gov/item/16002789/

Pinkerton, R. C. (1956). Information theory and melody. *Scientific American*, *194*(2), 77–87.

Allan, M. (2002). *Harmonising chorales in the style of johann sebastian bach* [Doctoral dissertation, Edinborough]. https://www.researchgate.net/publication/238242337_Harmonising_Chorales_in_the_Style_of_Johann_Sebastian_Bach

Altay, E. V., & Alatas, B. (2018). Music based metaheuristic methods for constrained optimization. *2018 6th International Symposium on Digital Forensic and Security (ISDFS 2018)*, 1–6. https://doi.org/10.1109/ISDFS.2018.8355355

Herremans, D., & Chew, E. (2019). Morpheus: Generating structured music with constrained patterns and tension [arXiv:1812.04832 [cs]]. *IEEE Transactions on Affective Computing*, *10*(4), 510–523. https://doi.org/10.1109/TAFFC.2017.2737984

Hild, H., Feulner, J., & Menzel, W. (1991). Harmonet: A neural net for harmonizing chorales in the style of l.s.bach. *Advances in Neural Information Processing Systems 4 (NIPS 1991)*.

Civit, M., Civit-Masot, J., Cuadrado, F., & Escalona, M. J. (2022). A systematic review of artificial intelligence-based music generation: Scope, applications, and future trends. *Expert Systems with Applications*, *209*, 118190. https://doi.org/10.1016/j.eswa.2022.118190

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks [arXiv:1409.3215 [cs]]. *Advances in Neural Information Processing Systems 27 (NIPS 2014)*. https://doi.org/10.48550/arXiv.1409.3215

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding [arXiv:1810.04805 [cs]]. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (ACL 2019)*. https://doi.org/10.48550/arXiv.1810.04805

Zeng, M., Tan, X., Wang, R., Ju, Z., Qin, T., & Liu, T.-Y. (2021). Musicbert: Symbolic music understanding with large-scale pre-training [arXiv:2106.05630 [cs]]. *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 791–800. https://doi.org/10.18653/v1/2021.findings-acl.70

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. https://api.semanticscholar.org/CorpusID:160025533

Liu, H., Chen, Z., Yuan, Y., Mei, X., Liu, X., Mandic, D., Wang, W., & Plumbley, M. D. (2023). Audioldm: Text-to-audio generation with latent diffusion models [arXiv:2301.12503 [cs]]. *Proceedings of the 40th International Conference on Machine Learning (PMLR)*. https://doi.org/10.48550/arXiv.2301.12503

Evans, Z., Parker, J. D., Carr, C. J., Zukowski, Z., Taylor, J., & Pons, J. (2024). Stable audio open [arXiv:2407.14358 [cs]]. *arXiv*, (arXiv:2407.14358). https://doi.org/10.48550/arXiv.2407.14358

Min, L., Jiang, J., Xia, G., & Zhao, J. (2023). Polyffusion: A diffusion model for polyphonic score generation with internal and external controls [arXiv:2307.10304 [cs]]. *24th Int. Society for Music Information Retrieval Conference (ISMIR 2023)*. http://arxiv.org/abs/2307.10304

Ji, S., Yang, X., & Luo, J. (2024). A survey on deep learning for symbolic music generation: Representations, algorithms, evaluations, and challenges. *ACM Computing Surveys*, 56(1), 1–39. https://doi.org/10.1145/3597493

Chen, H., Smith, J. B. L., Spijkervet, J., Wang, J.-C., Zou, P., Li, B., Kong, Q., & Du, X. (2024). Sympac: Scalable symbolic music generation with prompts and constraints [arXiv:2409.03055 [cs]]. *25th Int. Society for Music Information Retrieval Conference*. https://doi.org/10.48550/arXiv.2409.03055

Kingma, D. P., & Welling, M. (2014). Auto-encoding variational bayes [arXiv:1312.6114 [stat]]. *International Conference on Learning Representations (ICLR)*. https://doi.org/10.48550/arXiv.1312.6114

Défossez, A., Copet, J., Synnaeve, G., & Adi, Y. (2023). High fidelity neural audio compression [arXiv:2210.13438 [eess]]. *Transactions on Machine Learning Research*. https://doi.org/10.48550/arXiv.2210.13438

Fradet, N., Briot, J.-P., & Chhel, F. (2021). Miditok: A python package for midi file tokenization. *Extended Abstracts for the Late-Breaking Demo Session of the 22nd International Society for Music Information Retrieval Conference (ISMIR 2021)*.

Huang, Y.-S., & Yang, Y.-H. (2020). Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions. *Proceedings of the 28th ACM International Conference on Multimedia*, 1180–1188. https://doi.org/10.1145/3394171.3413671

Ryu, J., Dong, H.-W., Jung, J., & Jeong, D. (2024). Nested music transformer: Sequentially decoding compound tokens in symbolic music and audio generation [arXiv:2408.01180 [cs]]. *25th Int. Society for Music Information Retrieval Conference (ISMIR 2024)*. https://doi.org/10.48550/arXiv.2408.01180

Dong, H.-W., Chen, K., Dubnov, S., McAuley, J., & Berg-Kirkpatrick, T. (2023). Multitrack music transformer [arXiv:2207.06983]. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2023)*. https://doi.org/10.48550/arXiv.2207.06983

Sennrich, R., Haddow, B., & Birch, A. (2016). Neural machine translation of rare words with subword units [arXiv:1508.07909 [cs]]. *Association for Computational Linguistics (ACL 2016)*. https://doi.org/10.48550/arXiv.1508.07909

Van Kranenburg, P., Volk, A., & Wiering, F. (2013). A comparison between global and local features for computational classification of folk song melodies. *Journal of New Music Research*, 42(1), 1–18. https://doi.org/10.1080/09298215.2012.718790

Blacking, J. (1971). Deep and surface structures in venda music. *Yearbook of the International Folk Music Council*, 3, 91–108. https://doi.org/10.2307/767458

McKay, C. (2004). *Automatic genre classification using large high-level musical feature sets.* [Doctoral dissertation, McGill]. https://www.researchgate.net/publication/220723648_Automatic_Genre_Classification_Using_Large_High-Level_Musical_Feature_Sets

Rouard, S., Adi, Y., Copet, J., Roebel, A., & Défossez, A. (2024). Audio conditioning for music generation via discrete bottleneck features [arXiv:2407.12563]. *25th Int. Society for Music Information Retrieval Conference*. https://doi.org/10.48550/arXiv.2407.12563

Lu, P., Xu, X., Kang, C., Yu, B., Xing, C., Tan, X., & Bian, J. (2023). Musecoco: Generating symbolic music from text [arXiv:2306.00110 [cs]]. *arXiv*, (arXiv:2306.00110). https://doi.org/10.48550/arXiv.2306.00110

Tan, H., & Herremans, D. (2020). Music fadernets: Controllable music generation based on high-level features via low-level feature modelling. *Proc. of 21st International Society of Music Information Retrieval Conference, ISMIR 2020*. https://www.researchgate.net/publication/343500818_Music_FaderNets_Controllable_Music_Generation_Based_On_High-Level_Features_via_Low-Level_Feature_Modelling

Rütte, D. v., Biggio, L., Kilcher, Y., & Hofmann, T. (2023). Figaro: Generating symbolic music with fine-grained artistic control [arXiv:2201.10936]. *International Conference on Learning Representations*. https://doi.org/10.48550/arXiv.2201.10936

Wu, S.-L., Donahue, C., Watanabe, S., & Bryan, N. J. (2023). Music controlnet: Multiple time-varying controls for music generation [arXiv:2311.07069 [cs]]. *arXiv*, (arXiv:2311.07069). http://arxiv.org/abs/2311.07069

Lan, Y.-H., Hsiao, W.-Y., Cheng, H.-C., & Yang, Y.-H. (2024). Musicongen: Rhythm and chord control for transformer-based text-to-music generation [arXiv:2407.15060 [cs]]. *25th Int. Society for Music Information Retrieval Conference*. http://arxiv.org/abs/2407.15060

Lin, L., Xia, G., Jiang, J., & Zhang, Y. (2024). Content-based controls for music large language modeling [arXiv:2310.17162 [cs]]. *arXiv*, (arXiv:2310.17162). https://doi.org/10.48550/arXiv.2310.17162

Tal, O., Ziv, A., Gat, I., Kreuk, F., & Adi, Y. (2024). Joint audio and symbolic conditioning for temporally controlled text-to-music generation [arXiv:2406.10970 [cs]]. *25th Int. Society for Music Information Retrieval Conference*. http://arxiv.org/abs/2406.10970

Huang, Y., Ghatare, A., Liu, Y., Hu, Z., Zhang, Q., Sastry, C. S., Gururani, S., Oore, S., & Yue, Y. (2024). Symbolic music generation with non-differentiable rule guided diffusion [arXiv:2402.14285 [cs]]. *The Forty-First International Conference on Machine Learning*. https://doi.org/10.48550/arXiv.2402.14285

Wang, Z., Wang, D., Zhang, Y., & Xia, G. (2020). Learning interpretable representation for controllable polyphonic music generation [arXiv:2008.07122 [cs]]. *21st International Conference on Music Information Retrieval (ISMIR 2020)*. https://doi.org/10.48550/arXiv.2008.07122

Zhu, T., Liu, H., Jiang, Z., & Zheng, Z. (2024). Symbolic music generation with fine-grained interactive textural guidance [arXiv:2410.08435 [cs]]. *arXiv*, (arXiv:2410.08435). https://doi.org/10.48550/arXiv.2410.08435

Chew, E., Volk, A., & Lee, C.-Y. (2005). Dance music classification using inner metric analysis. In B. Golden, S. Raghavan, & E. Wasil (Eds.), *The next wave in computing, optimization, and decision technologies* (pp. 355–370). Springer US. https://doi.org/10.1007/0-387-23529-9_23

Haas, W. B., & Volk, A. (2016). Meter detection in symbolic music using inner metric analysis. *International Society for Music Information Retrieval Conference (ISMIR 2016)*. https://www.semanticscholar.org/paper/Meter-Detection-in-Symbolic-Music-Using-Inner-Haas-Volk/8a72fc0874e0476f3dc0b92a109ebf854a4551cc

Volk, A., Garbers, J., Van Kranenburg, P., Wiering, F., Grijp, L., & Veltkamp, R. C. (2009). Comparing computational approaches to rhythmic and melodic similarity in folksong research. In T. Klouche & T. Noll (Eds.), *Mathematics and computation in music* (pp. 78–87, Vol. 37). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-04579-0_8

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10684–10695. https://openaccess.thecvf.com/content/CVPR2022/html/Rombach_High-Resolution_Image_Synthesis_With_Latent_Diffusion_Models_CVPR_2022_paper.html

Lee, M., Doh, S., & Jeong, D. (2023). Annotator subjectivity in the musiccaps dataset. *HCMIR23: 2nd Workshop on Human-Centric Music Information Research)*, 1.

Shu, Y., Xu, H., Zhou, Z., Hengel, A. v. d., & Liu, L. (2024). Musebarcontrol: Enhancing fine-grained control in symbolic music generation through pre-training and counterfactual loss [arXiv:2407.04331 [cs]]. *arXiv*, (arXiv:2407.04331). https://doi.org/10.48550/arXiv.2407.04331

Koo, J., Wichern, G., Germain, F. G., Khurana, S., & Roux, J. L. (2024). Smitin: Self-monitored inference-time intervention for generative music transformers [arXiv:2404.02252]. *IEEE Open Journal of Signal Processing 2025*. https://doi.org/10.48550/arXiv.2404.02252

Zhang, L., Rao, A., & Agrawala, M. (2023). Adding conditional control to text-to-image diffusion models [arXiv:2302.05543 [cs]]. *IEEE/CVF International Conference on Computer Vision (ICCV 2023)*. https://doi.org/10.1109/ICCV51070.2023.00355

Xiong, Z., Wang, W., Yu, J., Lin, Y., & Wang, Z. (2023). A comprehensive survey for evaluation methodologies of ai-generated music [arXiv:2308.13736 [cs]]. *arXiv*, (arXiv:2308.13736). https://doi.org/10.48550/arXiv.2308.13736

Gurjar, K., & Moon, Y.-S. (2018). A comparative analysis of music similarity measures in music information retrieval systems. *Journal of Information Processing Systems*, *14*(1), 32–55. https://doi.org/10.3745/JIPS.04.0054

Kilgour, K., Zuluaga, M., Roblek, D., & Sharifi, M. (2019). Fréchet audio distance: A metric for evaluating music enhancement algorithms [arXiv:1812.08466 [eess]]. *INTERSPEECH 2019*. https://doi.org/10.48550/arXiv.1812.08466

Elizalde, B., Deshmukh, S., Ismail, M. A., & Wang, H. (2023). Clap: Learning audio concepts from natural language supervision [arXiv:2206.04769 [cs]]. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2023)*. https://doi.org/10.48550/arXiv.2206.04769

Yu, B., Lu, P., Wang, R., Hu, W., Tan, X., Ye, W., Zhang, S., Qin, T., & Liu, T.-Y. (2022). Museformer: Transformer with fine- and coarse-grained attention for music generation [arXiv:2210.10349 [cs]]. *Thirty-sixth Conference on Neural Information Processing Systems (NeurIPS 2022)*. https://doi.org/10.48550/arXiv.2210.10349

Raffel, C. (2016). *Learning-based methods for comparing sequences, with applications to audio-to-midi alignment and matching* [Doctoral dissertation, Columbia University].

Canonical jsb dataset chorals 389. (n.d.). https://github.com/czhuang/JSB-Chorales-dataset

Keith, J. (n.d.). Thesession data. https://github.com/adactio/TheSession-data

Hawthorne, C., Stasyuk, A., Roberts, A., Simon, I., Huang, C.-Z. A., Dieleman, S., Elsen, E., Engel, J., & Eck, D. (2019). Enabling factorized piano music modeling and generation with the MAESTRO dataset. *International Conference on Learning Representations*. https://openreview.net/forum?id=r1lYRjC9F7

Hook theory collection of pop-song transcriptions. (n.d.). https://www.hooktheory.com/theorytab

Wang, Z., Chen, K., Jiang, J., Zhang, Y., Xu, M., Dai, S., Gu, X., & Xia, G. (2020). Pop909: A pop-song dataset for music arrangement generation. *21st International Conference on Music Information Retrieval (ISMIR 2020)*.

Crestel, L., Esling, P., Heng, L., & McAdams, S. (2018). A database linking piano and orchestral midi scores with application to automatic projective orchestration [arXiv:1810.08611 [cs]]. *18th International Society for Music Information Retrieval Conference*. https://doi.org/10.48550/arXiv.1810.08611

Classical archives dataset donation. (n.d.). https://www.classicalarchives.com/newca/#!/

Bit midi dataset donation. (n.d.). https://bitmidi.com/

Ens, J., & Pasquier, P. (2020). Mmm: Exploring conditional multi-track music generation with the transformer [arXiv:2008.06048 [cs]]. *arXiv*, (arXiv:2008.06048). https://doi.org/10.48550/arXiv.2008.06048

Bertin-Mahieux, T., Ellis, D. P., Whitman, B., & Lamere, P. (2011). The million song dataset. *Proceedings of the 12th international conference on music information retrieval (ISMIR 2011)*.

4. Appendix

4.1 Comparing Tokenization lengths

Pretokenized Sequence Lengths - Mean: 47976.146146146144, Median: 43692.0, Std Dev: 23159.918513030065
Tokenized Sequence Lengths - Mean: 14519.587587587588, Median: 13159.0, Std Dev: 7197.064427699848

Applying the BPE tokenizer results in sequence lengths roughly 1/3 of the original sequences. The sample is taken from 1000 random music pieces from the lakh-midi dataset.
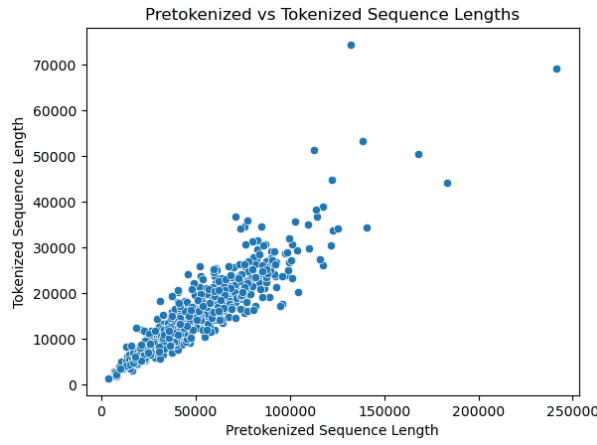

Figure 4.1: Scatterplot of Pre-tokenization vs bpe-tokenization sequence lengths

Name	Architecture	Control	FG	MI	Dataset
DeepBach (2017) Hadjeres et al., 2017	RNN	Inpainting	Yes	Yes	"Canonica
FolkRNN (2015) Sturm and Ben-Tal, 2016	RNN	meter, mode	No	No	Keith, n.d.
MT (2018) C.-Z. A. Huang et al., 2018	Transformer	-	No	No	Hawthorn
MidiNet (2019) Yang et al., 2017	GAN	chords, melody	Yes	Yes	"Hook The
Polyfussion 2023 Min et al., 2023	Diffusion	inpainting),texture	Yes	No	Wang, Che
FIGARO 2023 Rütte et al., 2023	Transformer	chords, instr, meter, nd	Yes	Yes	Raffel, 201
MMT 2023 Dong et al., 2023	Transformer	instrumentation	No	Yes	Raffel, 201
OMT 2020 Y.-S. Huang and Yang, 2020	Transformer	chords, tempo	Yes	No	C
MuseNet 2019 Christine, 2019	Transformer	instr, genre	Yes	Yes	Hawthorn
MMM 2020 Ens and Pasquier, 2020	Transformer	inpainting, instr, nd	Yes	Yes	Raffel, 201
Sympack 2024 Chen et al., 2024	Transformer	chords, structure, notes	Yes	Yes	Raffel, 201
MuseCoco 2023 Lu et al., 2023	Transformer	various	No	No	C
MBC 2024 Shu et al., 2024	Transformer	chords	Yes	No	Wang, Che
Museformer 2022 Yu et al., 2022	Transformer	-	No	Yes	Raffel, 201
NTT Ryu et al., 2024	Transformer	-	No	Yes	Raffel, 201
FTG Zhu et al., 2024	Diffusion	texture, rhythm, chords	Yes	No	Wang, Che
Fader Nets	VAE	rhythm, arousal	No	No	Hawthorn
NDRD Y. Huang et al., 2024	Diffusion	chord, pitch, nd	Yes	No	Hawthorn

Table 4.1: Overview of music generation models, their architectures, and control mechanisms. Empty columns indicate missing information on fine-grained control, multitrack capabilities, dataset, and evaluation methods.

4.2 Notes Comparing Symbolic Music Generators and their evaluation methods

- DeepBach Hadjeres et al., 2017 - RNN - inpainting. Evaluation Turing

- FolkRNN Sturm and Ben-Tal, 2016 - RNN - control for meter and mode - Expert Evaluation + Performance Practice

- MusicTransformer C.-Z. A. Huang et al., 2018 - Transformer - Evaluation: Subjective - Tournament Style between different generated and natural music. Objective: Validation NLL

- MidiNetYang et al., 2017 - GAN - Control for Chords/Priming melody. Human (how pleasing, how real, and how interesting)

- Polyfussion Min et al., 2023 - Diffusion Model - supports inpainting, interpolation, melody/accompanim generation, control for chord progression, texture. Subjective Evaluation Questionnaire for naturalness, creativity, musicality. Objective Control success.

- FIGARO Rütte et al., 2023 - Transformer Model - bar-wise control for chords, instrumentation, time-signature, note-density, mean-pitch. Evaluation: Perplexity (improvement over NLL for sequences of different length), Discription Fidelity (i.e accuracy in regards binary controls instruments, chords, time-signature). Macro Overlapping Area - comparison of feature histograms. Normalized Mean Root Square Area for note-density. Cosine similarity for chroma (melodic) and groove (rhythmic) feature vectors. Ablation study - effect of turning off controls. Extensive Subjective evaluation: 7569 comparisons by 691 participants - tournament style.

- Multi Track Music Transformer (MMT) Dong et al., 2023 - Transformer - Instrument control. Comparison of different tokenisation techniques, REMI+ and Compound Tokens. Compound tokens are more condensed and the generated samples are longer, achieves significant speedups and reduces memory usage (2.6 * MMM, or 3.5 * REMI+). Objective evaluations: Inference time, pitch class entropy, scale consistency, groove consistency, Human evaluation (90 comparisons by 9 participants) on Coherence Richness Arrangement Overall.

  Additional: Analysis of self attention as explanation avenue, which notes are most important.

- REMI pop music transformer - Y.-S. Huang and Yang, 2020 - Transformer - continuation, control local control over chord and tempo

- MuseNet Christine, 2019 - Transformer -,Instrument control, style control. No evaluation, only showcase.

- MMM Ens and Pasquier, 2020 - Transformer model - inpainting, instrument control, note-density. Introduce novel representation No Rigorous Evaluation

- SymPAC Chen et al., 2024 - Transformer - Control for Chords, structure, instrumentation, single notes. Train with both symbolic and transcribed audio data. Fine grained control. Constrained generation with a Finite State Machine. Comparison of three separate models trained on three datasets. **Evaluation** of controlability with KL-divergence on different controlled features over chords, structure, and individual notes. - KL divergence decreases with dataset size. 800 samples are generated and compared against a validation set of 3000 songs. Subjective evaluation (12 expert participants MIR researchers and music producers) on parameters of Coherence Richness Arrangement Structure

- MuseCoco Lu et al., 2023 - Transformer - Control via text for following attributes instru-

mentation, ambitus, rhythm (intensity and "dancability"), number of bars, time signature, key, tempo, duration, artist, emotion, genre. All of these controls are global controls, a description is converted into a list of attributes which is then used for generation. Combination of many datasets. Creation of text descriptions from attribute list with data from the dataset, and ChatGPT. Evaluation: Objective - text to attribute list. Subjective evaluation 19 participants with at least basic music knowledge - questions to Musicality, Controlability (adherence of sample to music description), Overal Impression. Comparison with LLM generated music (with no special training for music generation)

- MBD Shu et al., 2024 Extension of MuseCoco for time varying chord controls. Counterfactual Loss and Auxiliary task training improve controllability.

- Museformer Yu et al., 2022 - Transformer - No controls. Goal improve long-term structure with fine and coarse attention. Captures structure well. Objective Evaluation: Perplexity: prediction accuracy of next token, Similarity Error the error between the similarity distribution of training data and generated music Subjective Evaluation 10 Participants Musicality, Long Term Structure, Short Term Structure. Ablation study - evaluate objective effect of coarse and fine-grained attention. + Case study and detailed look at model.

- NMT Ryu et al., 2024 - Transformer improve longterm structure and reduce sequence length through compound tokens. Application to both symbolic and audio tokens. Cross attention vs self-attention comparison Evaluation: FAD, CLAP, KL and NLL over audio tokens. NLL over symbolic tokens. Subjective Evaluation Coherence, Richness, Consistency, Overall 29 participants. 8 selected prompts, 4 different continuations with REMI, Compound Word and 2 NMT variations.

- FTG - Fine Grained Texture Control - Diffusion - control over texture, rhythm and chords.

- Fader NetsTan and Herremans, 2020 - VAE - Control over rhythm, arousal, Idea: develop a "fader" representing a high-level abstract feature i.e arousal. Arousal is disentangled using a VAE into lower level features (i.e rhythmic density).

  Evaluation of the influence of latent features is on generated (style transfer) music.: Consistency, Restrictiveness (one latent dimension does not influence other musical features), Linearity (linear change in latent feature - linear change in musical feature) Subjective listening test to indicate success of arousal shift -> 48 participants, evaluate agreement with arousal direction.

- NDRD Symbolic Music Generation with Non-Differentiable Rule Guided Diffusion. Guidance of diffusion sampling with non-differentiable rules.