7. SOTA Transformers for Gen Mus

Generative Algorithms for Sound and Music

Universitat Pompeu Fabra Barcelona

MTG Music Technology Group

Overview

1. The origins + problems
2. Museformer
3. MuPT
4. Tips for Transformer assignment

MUSIC TRANSFORMER: GENERATING MUSIC WITH LONG-TERM STRUCTURE

Cheng-Zhi Anna Huang* **Ashish Vaswani** **Jakob Uszkoreit** **Noam Shazeer**
Ian Simon **Curtis Hawthorne** **Andrew M. Dai** **Matthew D. Hoffman**
Monica Dinculescu **Douglas Eck**
Google Brain

ABSTRACT

Music relies heavily on repetition to build structure and meaning. Self-reference occurs on multiple timescales, from motifs to phrases to reusing of entire sections of music, such as in pieces with ABA structure. The Transformer (Vaswani et al., 2017), a sequence model based on self-attention, has achieved compelling results in many generation tasks that require maintaining long-range coherence. This suggests that self-attention might also be well-suited to modeling music. In musical composition and performance, however, relative timing is critically important. Existing approaches for representing relative positional information in the Transformer modulate attention based on pairwise distance (Shaw et al., 2018). This is impractical for long sequences such as musical compositions since their memory complexity for intermediate relative information is quadratic in the sequence length. We propose an algorithm that reduces their intermediate memory requirement to linear in the sequence length. This enables us to demonstrate that a Transformer with our modified relative attention mechanism can generate minute-long compositions (thousands of steps, four times the length modeled in Oore et al. (2018)) with compelling structure, generate continuations that coherently elaborate on a given motif, and in a seq2seq setup generate accompaniments conditioned on melodies[1]. We evaluate the Transformer with our relative attention mechanism on two datasets, JSB Chorales and Piano-e-Competition, and obtain state-of-the-art results on the latter.

1 INTRODUCTION

A musical piece often consists of recurring elements at various levels, from motifs to phrases to sections such as verse-chorus. To generate a coherent piece, a model needs to reference elements that came before, sometimes in the distant past, repeating, varying, and further developing them to create contrast and surprise. Intuitively, self-attention (Parikh et al., 2016) appears to be a good match for this task. Self-attention over its own previous outputs allows an autoregressive model to access

Music Transformer: Generating music with long-term structure

Cheng-Zhi Anna Huang[*] **Ashish Vaswani** **Jakob Uszkoreit** **Noam Shazeer**
Ian Simon **Curtis Hawthorne** **Andrew M. Dai** **Matthew D. Hoffman**
Monica Dinculescu **Douglas Eck**
Google Brain

Abstract

Music relies heavily on repetition to build structure and meaning. Self-reference occurs on multiple timescales, from motifs to phrases to reusing of entire sections of music, such as in pieces with ABA structure. The Transformer (Vaswani et al., 2017), a sequence model based on self-attention, has achieved compelling results in many generation tasks that require maintaining long-range coherence. This suggests that self-attention might also be well-suited to modeling music. In musical composition and performance, however, relative timing is critically important. Existing approaches for representing relative positional information in the Transformer modulate attention based on pairwise distance (Shaw et al., 2018). This is impractical for long sequences such as musical compositions since their memory complexity for intermediate relative information is quadratic in the sequence length. We propose an algorithm that reduces their intermediate memory requirement to linear in the sequence length. This enables us to demonstrate that a Transformer with our modified relative attention mechanism can generate minute-long compositions (thousands of steps, four times the length modeled in Oore et al. (2018)) with compelling structure, generate continuations that coherently elaborate on a given motif, and in a seq2seq setup generate accompaniments conditioned on melodies[1]. We evaluate the Transformer with our relative attention mechanism on two datasets, JSB Chorales and Piano-e-Competition, and obtain state-of-the-art results on the latter.

1 Introduction

A musical piece often consists of recurring elements at various levels, from motifs to phrases to sections such as verse-chorus. To generate a coherent piece, a model needs to reference elements that came before, sometimes in the distant past, repeating, varying, and further developing them to create contrast and surprise. Intuitively, self-attention (Parikh et al., 2016) appears to be a good match for this task. Self-attention over its own previous outputs allows an autoregressive model to access

Music Transformer (2018)

- First transformer for gen mus
- Vanilla + relative attention
- Trained on piano (MIDI)
- Demonstrate attention works

Music Transformer: The good and the bad

Works locally

Incoherent long form

Music Transformer composer brain

Two solutions

- Encode music knowledge -> Museformer
- Fuc*k it, I’m going to brute-force it with scale -> MuPT

Why Transformers struggle

- Music sequences are long: 10k-20k+ tokens per song
- Music is not uniformly structured
    - Repetition & variation
    - Long-range dependencies at bar / phrase level

Time and memory in self-attention

$$Z(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V$$


WHEN YOU DISCOVER
ATTENTION SCALES QUADRATICALLY
imgflip.com

Long-sequence transformers

Long-sequence transformers

- Local attention / sliding windows
  - Miss long-distance repetitions

Long-sequence transformers

- Local attention / sliding windows
  - Miss long-distance repetitions
- Linear / approximate global attention
  - Lose precise correlations

Long-sequence transformers

- Local attention / sliding windows
  - Miss long-distance repetitions
- Linear / approximate global attention
  - Lose precise correlations
- Recurrent Transformers
  - Fixed memory, misaligned with musical form

What if we combine?

- Local attention / sliding windows
  - Miss long-distance repetitions
- Linear / approximate global attention
  - Lose precise correlations
- Recurrent Transformers
  - Fixed memory, misaligned with musical form

Museformer: Transformer with Fine- and Coarse-Grained Attention for Music Generation

Botao Yu[†], **Peiling Lu**[‡], **Rui Wang**[‡], **Wei Hu**[†*], **Xu Tan**[‡*],
Wei Ye[§], **Shikun Zhang**[§], **Tao Qin**[‡], **Tie-Yan Liu**[‡]
[†]State Key Laboratory for Novel Software Technology, Nanjing University, China
[‡]Microsoft Research Asia
[§]National Engineering Research Center for Software Engineering, Peking University, China
btyu@foxmail.com, {peil,ruiwa,xuta,taoqin,tyliu}@microsoft.com,
whu@nju.edu.cn, {wye,zhangsk}@pku.edu.cn

https://github.com/microsoft/muzic

Abstract

Symbolic music generation aims to generate music scores automatically. A recent trend is to use Transformer or its variants in music generation, which is, however, suboptimal, because the full attention cannot efficiently model the typically long music sequences (e.g., over 10,000 tokens), and the existing models have shortcomings in generating musical repetition structures. In this paper, we propose Museformer, a Transformer with a novel fine- and coarse-grained attention for music generation. Specifically, with the fine-grained attention, a token of a

Museformer: Transformer with Fine- and Coarse-Grained Attention for Music Generation

Botao Yu[†], **Peiling Lu**[‡], **Rui Wang**[‡], **Wei Hu**[†*], **Xu Tan**[‡*],
Wei Ye[§], **Shikun Zhang**[§], **Tao Qin**[‡], **Tie-Yan Liu**[‡]

[†]State Key Laboratory for Novel Software Technology, Nanjing University, China
[‡]Microsoft Research Asia
[§]National Engineering Research Center for Software Engineering, Peking University, China
btyu@foxmail.com, {peil,ru[illegible],[illegible]oqin,tyliu}@microsoft.com,
whu@[illegible]ju.ed[illegible]cn, {[illegible]zhangsk}@pku.edu.cn

[illegible]s://g[illegible]thub.com/microsoft/muzic

Abstract

Symbolic music generation aims to generate music scores automatically. A recent trend is to use Transformer or its variants in music generation, which is, however, suboptimal, because the full attention cannot efficiently model the typically long music sequences (e.g., over 10,000 tokens), and the existing models have shortcomings in generating musical repetition structures. In this paper, we propose Museformer, a Transformer with a novel fine- and coarse-grained attention for music generation. Specifically, with the fine-grained attention, a token of a

Core idea: Two kinds of attention

- Fine-grained attention
  - Exact token-level attention
  - Applied to current bar + structure-related bars
  - Captures details

Core idea: Two kinds of attention

- Fine-grained attention
  - Exact token-level attention
  - Applied to current bar + structure-related bars
  - Captures details
- Coarse-grained attention
  - Approximate attention via summaries
  - Applied to all other bars
  - Captures approximation

Architecture

- Decoder-only autoregressive Transformer
- Replace self-attention with Fine- & Coarse-Grained Attention (FC-Attention)
- 16M parameters

Representation

Representation

- REMI-like
- Music tokens grouped into bars
- Summary token after each bar
- ~160 tokens / bar

FC-Attention: 2-step process

1. Summarization
    - Compress each bar into one vector
    - Preserve musically relevant information
    - Enable cheap global context later (coarse)

FC-Attention: 2-step process

1. Summarization
    - Compress each bar into one vector
    - Preserve musically relevant information
    - Enable cheap global context later (coarse)
2. Aggregation
    - Use exact detail where structure matters (fine-grained)
    - Use summaries elsewhere

Summarization

$$\tilde{\boldsymbol{s}}_i = \mathrm{Attn}\big(\boldsymbol{s}_i, [\boldsymbol{X}_i, \boldsymbol{s}_i]\big)$$

Summarization

$$\tilde{\boldsymbol{s}}_i = \mathrm{Attn}(\boxed{\boldsymbol{s}_i}, [\boldsymbol{X}_i, \boldsymbol{s}_i])$$

summary token / query

$\boldsymbol{s}_i \in \mathbb{R}^{1\times d}$

Summarization

$$\tilde{\boldsymbol{s}}_i = \mathrm{Attn}(\boxed{\boldsymbol{s}_i}, [\boxed{\boldsymbol{X}_i}, \boldsymbol{s}_i])$$

summary token / query

$\boldsymbol{s}_i \in \mathbb{R}^{1 \times d}$

tokens for bar *i*

$X_i = \{x_{i,1}, \ldots, x_{i,m_i}\}$

Summarization

$$\tilde{s}_i = \mathrm{Attn}(\boxed{\boldsymbol{s}_i}, \boxed{[\boxed{\boldsymbol{X}_i}, \boldsymbol{s}_i]})$$

summary token / query

$\boldsymbol{s}_i \in \mathbb{R}^{1\times d}$

tokens for bar *i*

$X_i = \{x_{i,1}, \dots, x_{i,m_i}\}$

keys / values

$[X_i, \boldsymbol{s}_i] \in \mathbb{R}^{(m_i+1)\times d}$

Summarization

$$\boxed{\tilde{\boldsymbol{s}}_i} = \mathrm{Attn}\left(\boxed{\boldsymbol{s}_i}, \boxed{[\boxed{\boldsymbol{X}_i}, \boldsymbol{s}_i]}\right)$$

bar representation

$\tilde{\boldsymbol{s}}_i \in \mathbb{R}^{1\times d}$

summary token / query

$\boldsymbol{s}_i \in \mathbb{R}^{1\times d}$

tokens for bar *i*

$X_i = \{x_{i,1}, \ldots, x_{i,m_i}\}$

keys / values

$[X_i, \boldsymbol{s}_i] \in \mathbb{R}^{(m_i+1)\times d}$

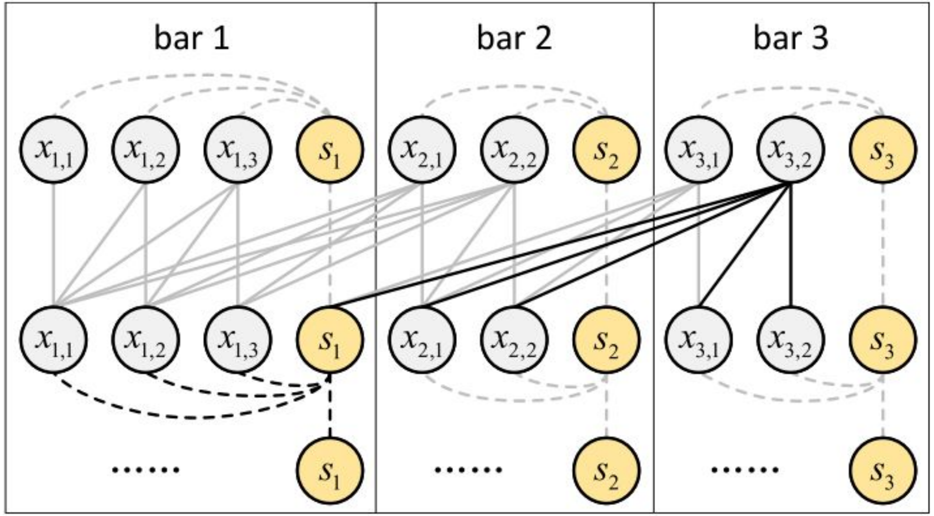
Summarization: Interpretation

- Compressor
- Learned representation of 1 bar
- Different tokens weighted differently
- Happens independently for each bar

$$\tilde{\boldsymbol{s}}_i = \mathrm{Attn}\big(\boldsymbol{s}_i, [\boldsymbol{X}_i, \boldsymbol{s}_i]\big)$$

Aggregation

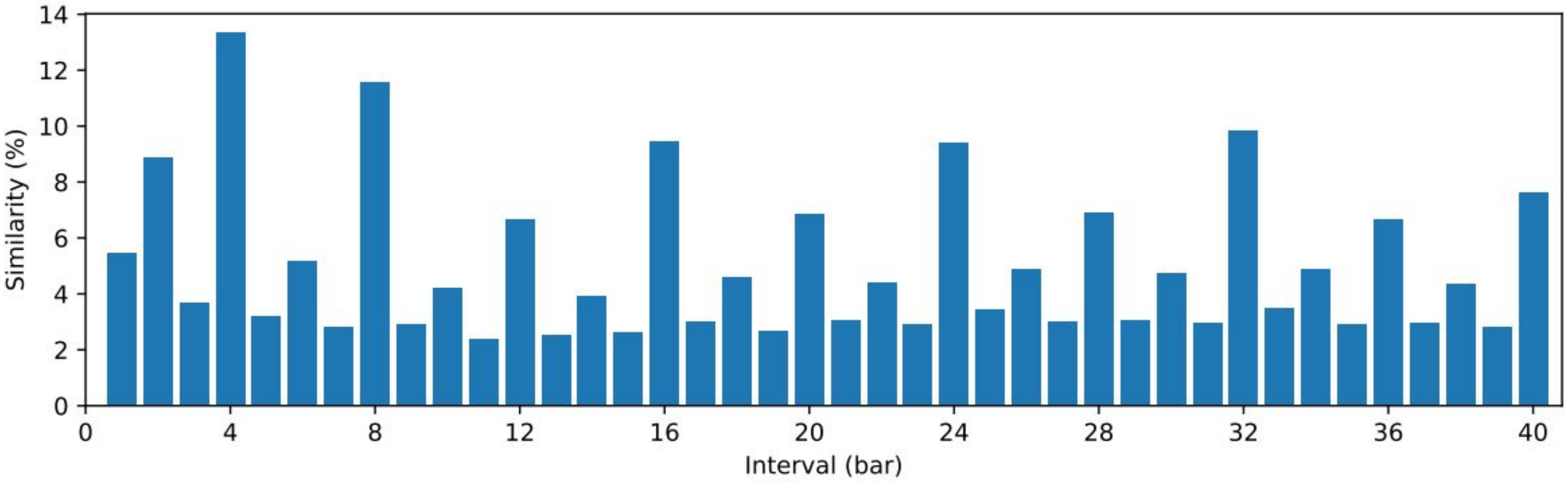
- When predicting token $x_{i,j}$:
  - Use exact detail where structure matters
  - Use summaries elsewhere



Structure-related bars

- Selected using similarity statistics on human music
- Bars tend to repeat at distances:
  - 1, 2, 4, 8, 16, 24 32, …
- Fixed set of 8 structure-related bars per bar

Structure-related bars



Aggregation

$$\tilde{\boldsymbol{x}}_{i,j} = \mathrm{Attn}\Big(\boldsymbol{x}_{i,j}, [\boldsymbol{X}_{R(i)}, \boldsymbol{X}_{i,k \leq j}, \tilde{\boldsymbol{S}}_{\bar{R}(i)}]\Big)$$

Aggregation

$$\tilde{\boldsymbol{x}}_{i,j} = \mathrm{Attn}\left(\boldsymbol{x}_{i,j}, [\boxed{\boldsymbol{X}_{R(i)}}, \boxed{\boldsymbol{X}_{i,k\leq j}}, \boxed{\tilde{\boldsymbol{S}}_{\bar{R}(i)}}]\right)$$

tokens from structure related bars

previous tokens in current bar

summaries from previous bars

Aggregation

fine-grained

coarse

$$\tilde{\boldsymbol{x}}_{i,j} = \text{Attn}\left(\boldsymbol{x}_{i,j}, [\boxed{\boxed{\boldsymbol{X}_{R(i)}}, \boxed{\boldsymbol{X}_{i,k\leq j}}}; \boxed{\tilde{\boldsymbol{S}}_{\bar{R}(i)}}]\right)$$

tokens from structure related bars

previous tokens in current bar

summaries from previous bars

Aggregation

$$\tilde{\boldsymbol{x}}_{i,j} = \mathrm{Attn}\Big(\boxed{\boldsymbol{x}_{i,j}}, \boxed{[\boldsymbol{X}_{R(i)}, \boldsymbol{X}_{i,k\leq j}, \tilde{\boldsymbol{S}}_{\bar{R}(i)}]}\Big)$$

query

key / value

Aggregation

$$\tilde{\boldsymbol{x}}_{i,j} = \mathrm{Attn}\left(\boxed{\boldsymbol{x}_{i,j}}, \boxed{[\boldsymbol{X}_{R(i)}, \boldsymbol{X}_{i,k\leq j}, \tilde{\boldsymbol{S}}_{\bar{R}(i)}]}\right)$$

query

key / value

$$\text{context} = \begin{bmatrix} \text{important tokens} \\ \text{local tokens} \\ \text{summaries} \end{bmatrix}$$

Aggregation: Interpretation

- Detail via structurally important tokens
- Global context retained via summaries
- Independent for each token
- Repetitions & motifs

$$\tilde{\boldsymbol{x}}_{i,j} = \mathrm{Attn}\Big(\boldsymbol{x}_{i,j}, [\boldsymbol{X}_{R(i)}, \boldsymbol{X}_{i,k\leq j}, \tilde{\boldsymbol{S}}_{\bar{R}(i)}]\Big)$$

Instead of each token attending to all keys and values, each token attends to a small, musically meaningful K/V set.

Training

- Lakh MIDI dataset
- 30K songs
- 1,700 hours
- 95 bars / song

Evaluation

	Musicality	ST structure	LT structure	Overall	Pref
Music Transformer	6.00 ± 2.21	6.90 ± 1.76	5.30 ± 2.58	5.90 ± 1.90	0.20
Transformer-XL	6.10 ± 2.19	7.40 ± 1.81	6.26 ± 2.78	6.44 ± 2.01	0.34
Longformer	6.46 ± 1.81	7.60 ± 1.47	6.18 ± 2.54	6.44 ± 1.72	0.24
Linear Transformer	6.06 ± 1.99	6.92 ± 2.03	5.78 ± 2.64	6.30 ± 1.84	0.24
Museformer (ours)	$\mathbf{6.88 \pm 1.95}$	$\mathbf{7.86 \pm 1.51}$	$\mathbf{6.72 \pm 2.74}$	$\mathbf{7.12 \pm 1.81}$	**0.46**

Why Museformer (kind of) works

- Injects knowledge about form in design, via FC-Attention
- Avoids uniform approximation

What works? What are the limitations?

https://ai-muzic.github.io/museformer/

MuPT: A Generative Symbolic Music Pretrained Transformer

Xingwei Qu[1,3,4*], **Yuelin Bai**[5*], **Yinghao Ma**[1,7*],
Ziya Zhou[3], **Ka Man Lo**[3], **Jiaheng Liu**[1], **Ruibin Yuan**[1,3], **Lejun Min**[8], **Xueling Liu**[1],
Tianyu Zhang[9], **Xinrun Du**[1], **Shuyue Guo**[1], **Yiming Liang**[10], **Yizhi Li**[1,4], **Shangda Wu**[11],
Junting Zhou[12], **Tianyu Zheng**[1], **Ziyang Ma**[13], **Fengze Han**[1], **Wei Xue**[3], **Gus Xia**[8],
Emmanouil Benetos[7], **Xiang Yue**[1], **Chenghua Lin**[4], **Xu Tan**[14], **Stephen W. Huang**[15]
Jie Fu[3†], **Ge Zhang**[1,2,6*†]

[1]M-A-P, [2]University of Waterloo, [3]HKUST, [4]University of Manchester,
[5]Shenzhen Institute of Advanced Technology, CAS, [6]Vector Institue, [7]QMUL, [8]MBZUAI,
[9]MILA, [10]Institute of Automation, CAS, [11]Central Conservatory of Music,
[12]PKU, [13]SJTU, [14]MSRA, [15]harmony.ai

https://map-mupt.github.io/

Abstract

In this paper, we explore the application of Large Language Models (LLMs) to the pre-training of music. While the prevalent use of MIDI in music modeling is well-established, our findings suggest that LLMs are inherently more compatible with ABC Notation, which aligns more closely with their design and strengths, thereby enhancing the model's performance in musical composition. To address the challenges associated with misaligned measures from different tracks during generation, we propose the development of a Synchronized Multi-Track ABC Notation (**SMT-ABC Notation**), which aims to preserve coherence across multiple musical tracks. Our contributions include a series of models capable of handling up to 8192 tokens, covering 90% of the symbolic music data in our training set. Furthermore, we explore the implications of the Symbolic Music Scaling Law (**SMS Law**) on model performance. The results indicate a promising direction for future research in music generation, offering extensive resources for community-led research through our open-source contributions.

1 Introduction

Large Language Models (LLMs) have experienced remarkable advancements, leading to their broad application across numerous domains. As these models extend into multimodal areas, such as visual and auditory fields, their capability to represent and model complex information, including images (Liu et al., 2023) and speech (Baevski et al., 2020) becomes

MuPT: Philosophy

Treat music like language

Music Pretrained Transformer (MuPT, 2024)

- GPT-style foundation model for symbolic music
- Trained at LLM scale
- ABC notation

Why not MIDI?

- Extremely long sequences
- Performance-level noise
- Weak explicit structure (bars, repeats, form)

Why ABC notation?

- Textual, compact, and hierarchical
- Explicitly encodes bars, repetition, sections
- Aligns well with next-token prediction

However…

- Standard ABC notation encodes voices sequentially:
    - Voice 1: bar 1, bar 2, bar 3, …
    - Voice 2: bar 1, bar 2, bar 3, …
- When trained autoregressively, this causes:
    - Bar misalignment across voices
    - Weak harmonic coordination

The solution: Synchronized Multi-Track ABC (SMT-ABC)

Reorder ABC notation by bar index, not by voice:

- Bar 1 from all tracks grouped together
- Bar 2 from all tracks grouped together
- Groups wrapped in a special <|> token

The solution: Synchronized Multi-Track ABC (SMT-ABC)

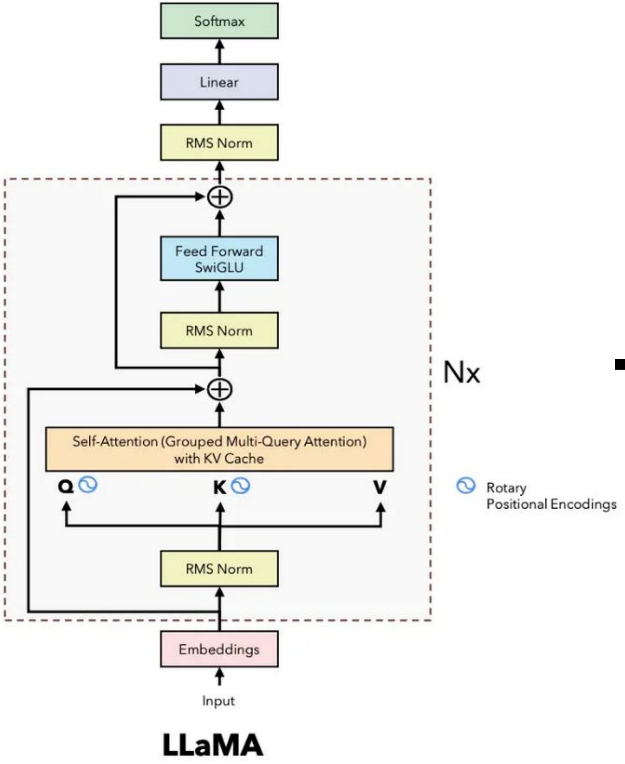
Tokenization

- BPE tokenizer (YouTokenToMe)
- Vocabulary size: 50k
- No normalization or artificial prefixes
- Explicit token for spaces <n>

Model architecture

- Decoder-only autoregressive Transformer
- Context length: 8192 tokens
- Sizes: 190M → 4.23B parameters
- Trained on 34B ABC tokens

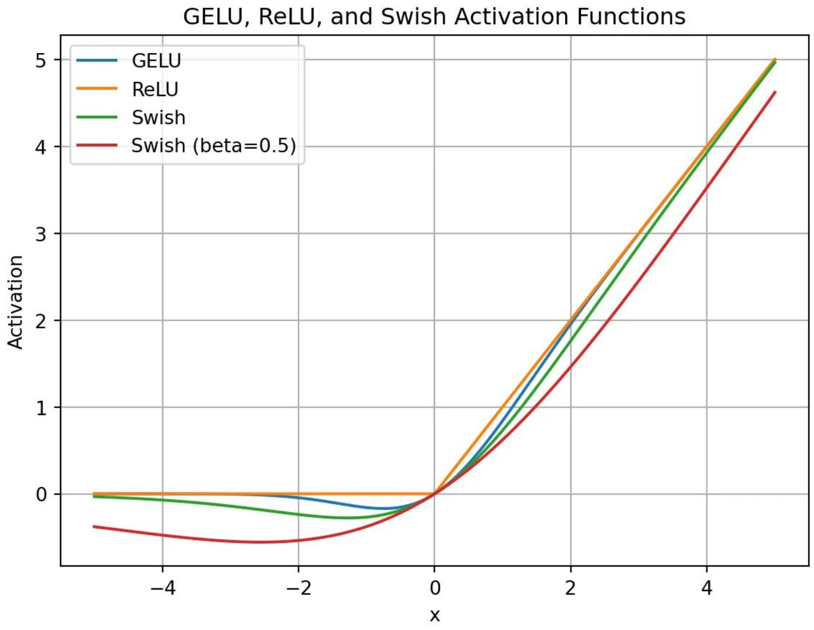
MuPT not-so-elegant recipe



+ sh*it ton computation + sh*it ton ABC data

Modern LLM ingredients

- Rotary Positional Encoding (RoPE)
- Root Mean Square Normalization
- SwiGLU

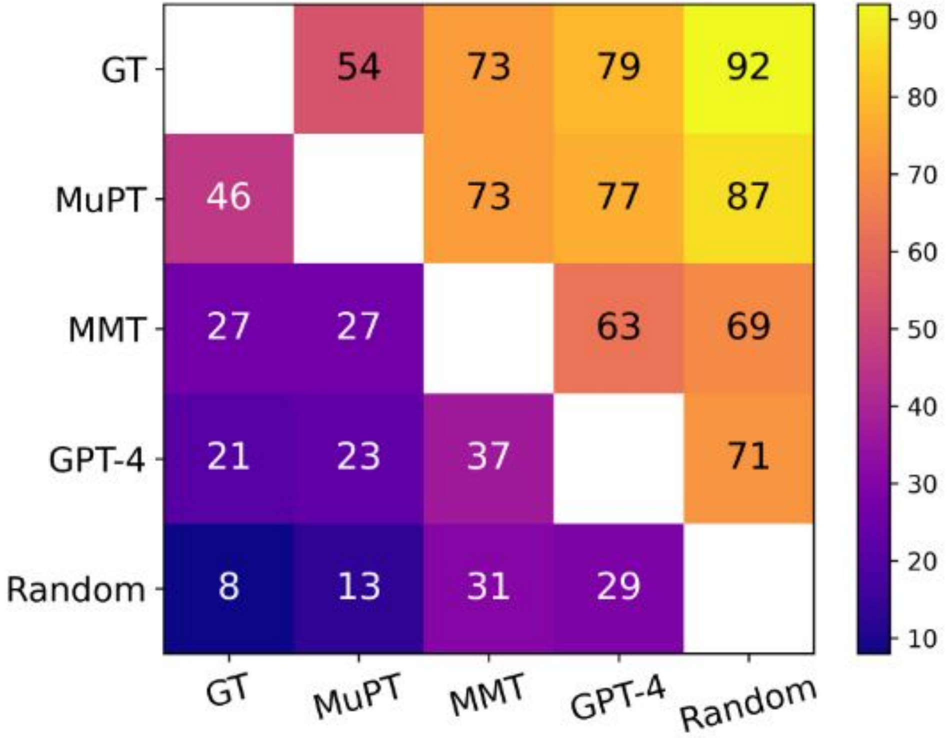


Evaluation: Objective

- Repetition rate close to human ground truth
- Higher intra-piece texture similarity

System	Texture similarity	Repetition Det. Rate (%)
MuPT	**0.4288**	**44.3**
GT	0.3729	43.5
MMT	0.1767	-
GPT-4	0.3614	16.9

Evaluation: A/B test



What works? What are the limitations?

https://map-mupt.github.io/

Museformer

- Solves long-range structure explicitly
- Uses hand-designed inductive bias:
    - Bar summaries
    - Fine vs coarse attention
    - Structure-related bars
- Works well at moderate scale
- Strong architectural prior about musical form

MuPT

- Solves long-range structure implicitly
- Relies on:
    - Better representation (ABC + SMT)
    - Much larger context (8k)
    - Massive pretraining
- No custom attention
- Treats music as a language modeling problem

From clever architectures

Boring architectures + massive scale

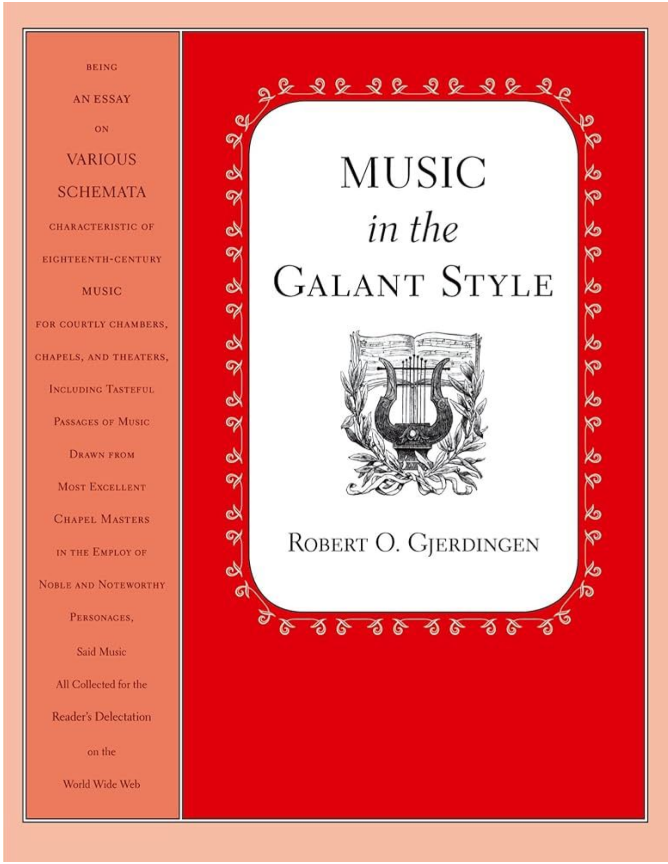
The more music knowledge encoded, the less:

1. dumb the model
2. scale needed

Is musical structure better encoded explicitly (Museformer) or learned implicitly (MuPT)?

Why?

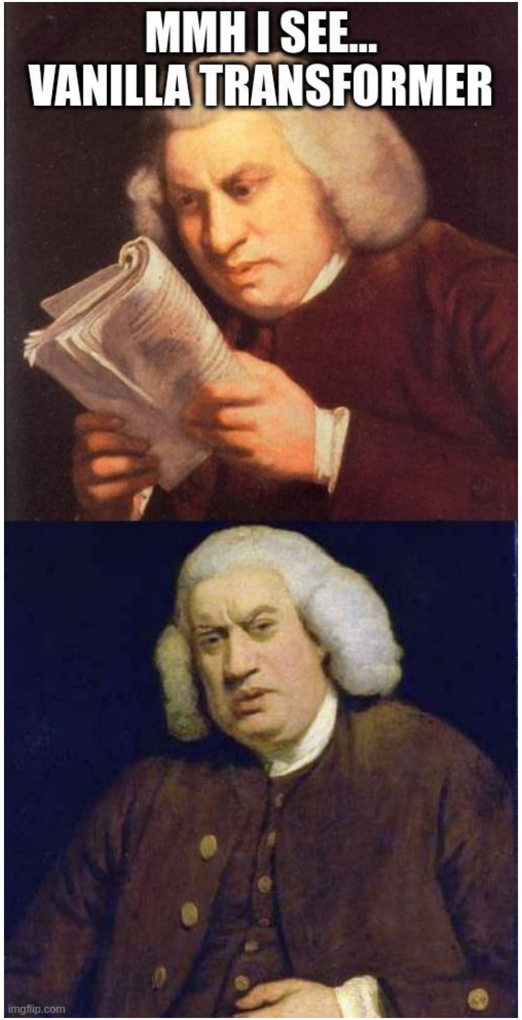
Music gestures



Bach Chorale Transformer

Approaches

- Fine-tune (HuggingFace)
- Re-adapt Valerio's code
- Implement / adapt paper
- Encoder - decoder or decoder only?
- New architecture?


MMH I SEE...
VANILLA TRANSFORMER
imgflip.com

Representation

- From MIDI or score?
- Structure / bar info?
- How do we represent SATB polyphony?
- What tokenizer do we use?

Heads Up: Hugging Face Class on Thursday by Fernando

Check website!