Music Generation by GPT Model

2023中研院實習生
資訊工程系 二年級 洪子涵

2023.08.31

Reference

- Ens, J., & Pasquier, P. (2020). MMM: Exploring Conditional Multi-Track Music Generation with the Transformer. arXiv:2008.06048. https://arxiv.org/abs/2008.06048
- Copet, J., Kreuk, F., Gat, I., Remez, T., Kant, D., Synnaeve, G., Adi, Y., & Défossez, A. (2023). Simple and Controllable Music Generation. arXiv:2306.05284.

  https://arxiv.org/abs/2306.05284

Overview

- Introduction
- Methodology
- Experiment

Music Representation

- Matrix (i.e. a piano roll)
- Sequence of tokens

Matrix Representation

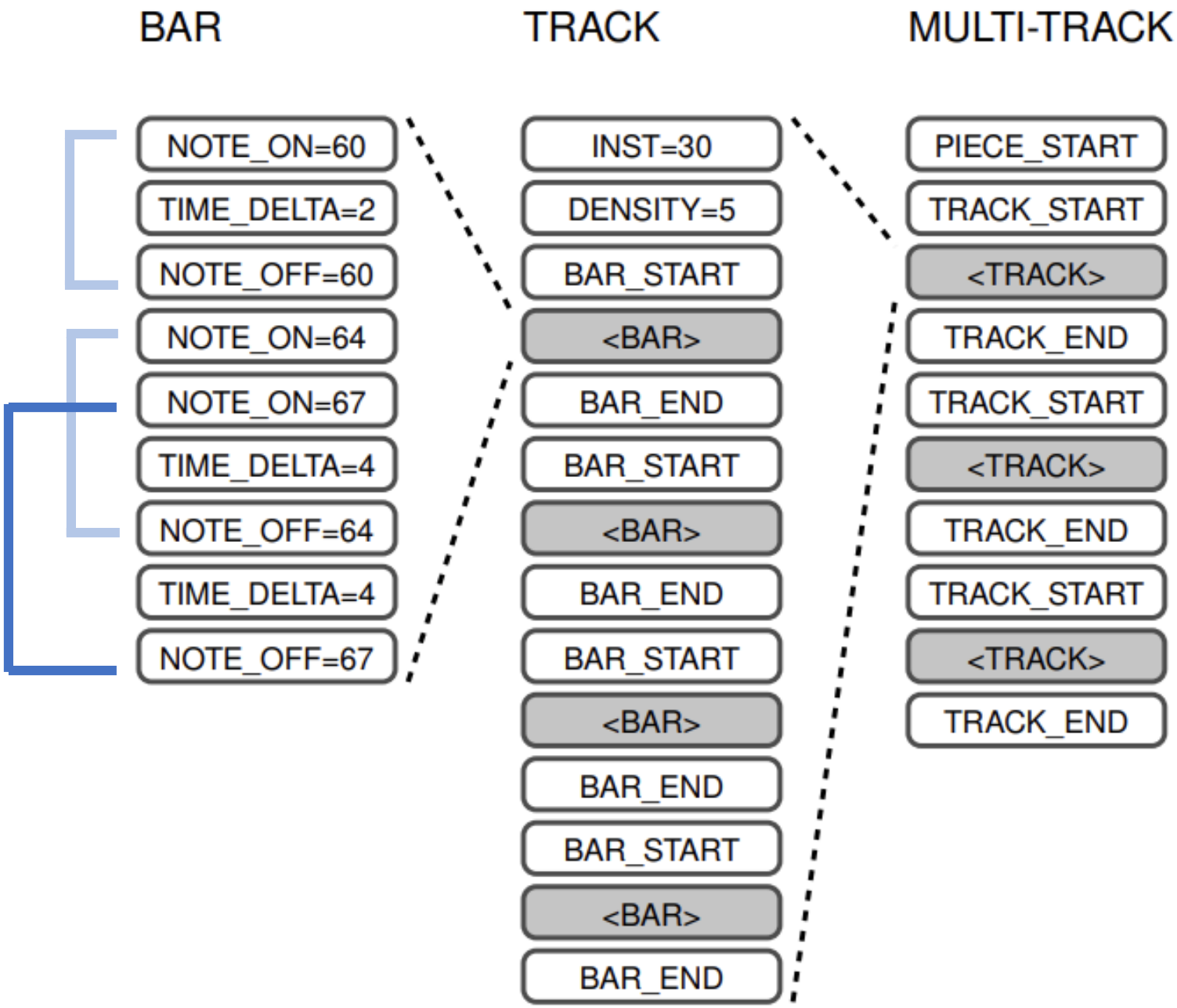
- A piano roll is a boolean matrix $x \in \{0, 1\}^{T \times P}$, where T is the number of time-steps and P is the number of pitches. Typically P = 128, allowing the piano roll to represent all possible MIDI pitches.
- Multi-track musical material can be represented using a boolean tensor $x \in \{0, 1\}^{M \times T \times P}$, where M is the number of tracks.
- However, using this type of representation is inherently inefficient, as the number of inputs increases by $T \times P$ for each track that is added, and accommodating small note lengths substantially increases T.

Sequence of tokens Representation

- Each token corresponds to a specific musical event or piece of metadata.
- 128 NOTE ON tokens, 128 NOTE OFF tokens, and 48 TIME SHIFT tokens. (Since musical events are quantized using 12 subdivisions per beat, 48 TIME SHIFT tokens allow for the representation of any rhythmic unit from sixteenth note triplets to a full 4-beat bar of silence.)
- Each bar begins with a BAR START token, and ends with a BAR END token.
- Tracks are simply a sequence of bars delimited by TRACK START and TRACK END tokens. Following the TRACK START token, an INSTRUMENT token is used to specify the MIDI program which is to be used to play the notes.

Sequence of tokens Representation

- Since there are 128 possible MIDI programs, we have 128 distinct INSTRUMENT tokens.
- A DENSITY LEVEL token follows the INSTRUMENT token, and indicates the note density of the current track.
- A piece is simply a sequence of tracks, however, all tracks sound simultaneously rather than being played one after the other.
- A piece begins with the PIECE START token.


BAR
TRACK
MULTI-TRACK
NOTE_ON=60
TIME_DELTA=2
NOTE_OFF=60
NOTE_ON=64
NOTE_ON=67
TIME_DELTA=4
NOTE_OFF=64
TIME_DELTA=4
NOTE_OFF=67
INST=30
DENSITY=5
BAR_START
<BAR>
BAR_END
BAR_START
<BAR>
BAR_END
BAR_START
<BAR>
BAR_END
BAR_START
<BAR>
BAR_END
PIECE_START
TRACK_START
<TRACK>
TRACK_END
TRACK_START
<TRACK>
TRACK_END
TRACK_START
<TRACK>
TRACK_END

MIDI File

- MIDI Files contain one or more MIDI streams, with time information for each event.
- Song, sequence, and track structures, tempo and time signature information, are all supported.
- Track names and other descriptive information may be stored with the MIDI data.

Why MIDI

- Discrete token
  - Easier to train
- Data size
  - Für Elise (02: 43)
  - MIDI (37KB) vs. MP3 (2561KB)
- Changeable instruments
- Data sources are easily accessible

MIDI Structure

- MIDI Files are composed of chunks.
- Each chunk has a 4-character type and a 32-bit length (number of bytes in the chunk).
- This structure allows easy addition of new chunk types that can be ignored by older programs.
- Each chunk starts with a 4-character ASCII type followed by a 32-bit length.
- The length is represented in big-endian format (e.g., 00 00 00 06 for a length of 6).
- The length refers to the data bytes that follow; the type and length bytes are not counted.
- A chunk's length includes 8 bytes for type and length, so a length of 6 occupies 14 bytes in the file.

Experiment Motivation

- Similarity Observation:
  - compare music with other input data to identify commonalities for model selection
- Sequence Representation:
  - both music and language are presented as continuous sequences displayed in chronological order
- Structure and Rules:
  - Language has grammar and structural rules, while music encompasses chords, melodies, compositional structures, etc.
- GPT Model:
  - By leveraging the model's contextual understanding and generation capabilities, the aim is to apply it to music generation.
- Expectation:
  - to preserve the overall coherence with a consistent style, avoiding significant discrepancies between different sections.

Dataset

- Mutopia Piano Dataset
  - 788 classical piano Pieces
  - vary lengths (about 70 hours in total)
  - https://www.mutopiaproject.org

Data Preprocessing

- Format Conversion:
  - Converting music Information from MIDI files into easily processable text Format.
- Tokenization:
  - Transforming the musical note sequences from music into text sequences, where each note and its associated information are mapped to a specific token.
- Vocabulary Creation:
  - Establishing a vocabulary based on the transformed tokens. This vocabulary serves as the foundation for training the GPT model.

Tokenizer

- TIME_SIGNATURE=4_4 is just one token.
  - Not 3 tokens TIME_SIGNATURE、=、4_4
  - Ex. 3_4、2_4 、 6_8 、 2_2 、 3_8 、 12_8 、 6_4

- White space split
- PIECE_START TIME_SIGNATURE=3_8 BPM=72 TRACK_START INST=0 DENSITY=4 BAR_START NOTE_ON=76 TIME_DELTA=1.0 NOTE_OFF=76 NOTE_ON=75 …

Model Description

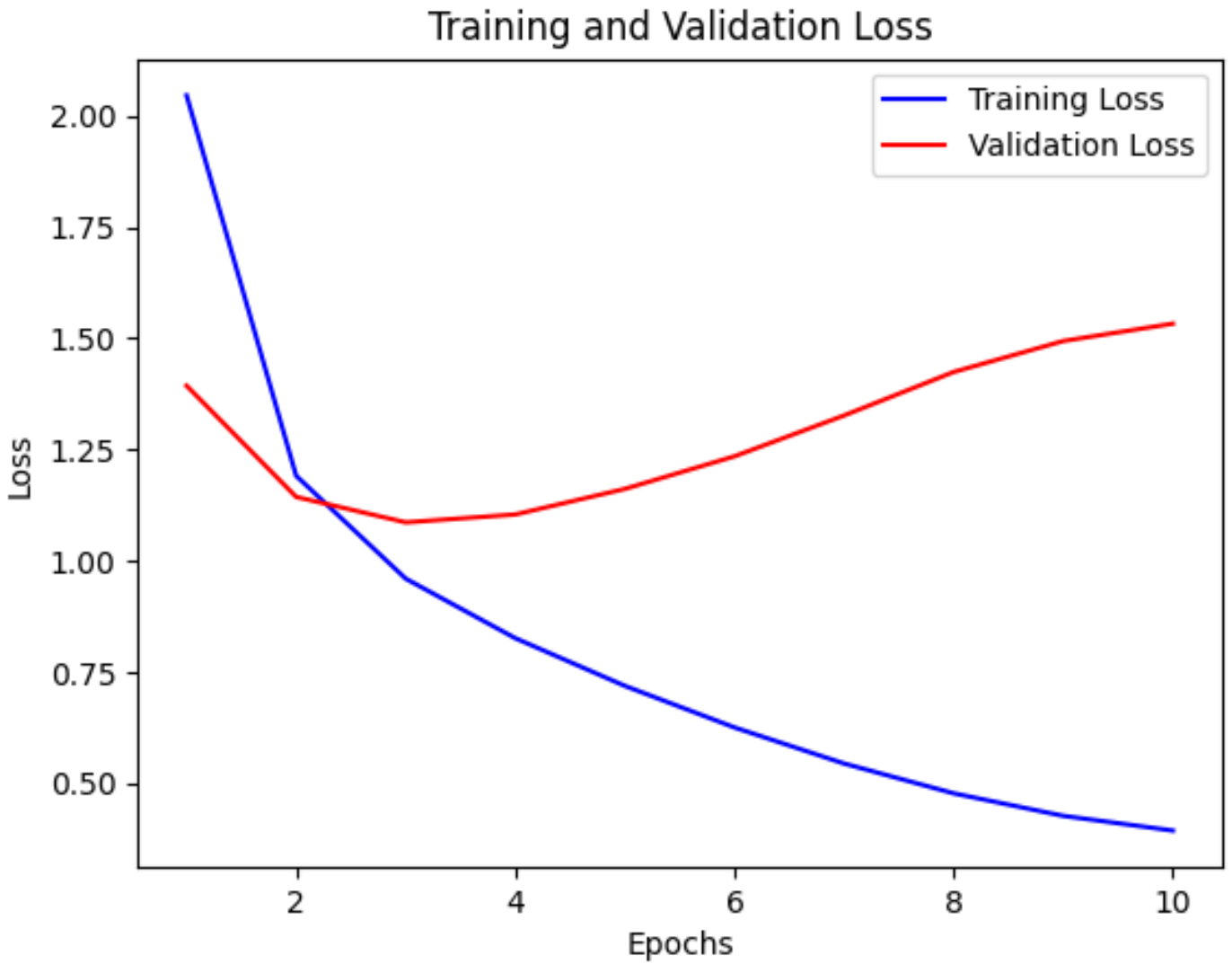
- Model architecture using GPT-2 Structure.
- The customized vocabulary built from the dataset is utilized. These tokens are designed to depict various musical attributes such as notes, instruments, and other music-related characteristics.
- In order for the GPT model to comprehend these custom musical data tokens, it’s better to train the model from scratch rather than fine-tuning it.

Training and Evaluation

- Utilize text files as input to train the model.
- Generate corresponding music sequence text.
- Monitor the change in loss as epochs progress.
- After each epoch, generate multiple music files using the same seed.
- Assess the quality of the music through subjective listening.

Progress

- Loss
- Epoch evolve



Model Output Generation

- Method 1: Free Generation Based on Music Information
  - Input specific information such as tempo and instruments in designated rhythms.
  - Use the provided tempo and other input information to freely generate corresponding music segments, considering musical structure and characteristics.
- Method 2: Sequential Generation Based on Specified Music Segments
  - Input user-provided music segments.
  - Generates subsequent music segments based on the input music segment.
  - This process is similar to how a general GPT model generates subsequent text based on preceding context.

Demo

- Random:
  - random\random_0.mid
  - random\random_1.mid
  - random\random_2.mid

Demo

- Information:
  - TIME_SIGNATURE=4_4
  - BPM=120
  - INST=0
  - information\epoch_1.mid
  - information\epoch_2.mid
  - information\epoch_5.mid
  - information\epoch_9.mid

Demo

Prefix:

- prefix\fur_Elise.mid
  - prefix\fur_Elise (1).mid
  - prefix\fur_Elise (2).mid

- prefix\moonlight.mid
  - prefix\moonlight (1).mid

Demo

- Different instrument:
  - different_instrument\inst_23.mid
    - Harmonica
  - different_instrument\inst_54.mid
    - Voice Oohs
- Different dataset (guitar):
  - different_dataset_guitar\guitar_1.mid
  - different_dataset_guitar\guitar_2.mid
  - different_dataset_guitar\guitar_7.mid
  - different_dataset_guitar\guitar_8.mid

Challenges and Possible Improvement:

- Assessing the quality of generated music is difficult to quantify, as musical quality is subjective. Subjective evaluation is time-consuming and varies from person to person.
  - Desire for more effective evaluation methods. These could include considering musical characteristics such as chord progressions and rhythm variations.
- Existing training data is influenced by composers' constrained thinking, considering human playability. This might result in the model generating music with similar limitations.
  - Hope to break through these constraints and possess greater creativity.

Thanks for Listening

Q & A