

Homework 4: Pop Music Transformer

20213013 Jiho Kang

December 14th, 2021

[Learning Process]

Favorite piano song as input for *Pop Music Transformer* is *Sonata in D Major, K. 118*, composed by *Domenico Scarlatti*. It was transcribed using the ONF model implemented in HW3 and created as a midi file. The transcribed midi file has the same velocity (can be seen as perceptual loudness [1]) as 127. If this is used as it is, an error ('KeyError: Note Velocity_31') occurs during the generation process, so a random value between 20 and 60 is assigned as shown in Figure 1. Figure 2 illustrates before and after the velocity is randomly changed. The data (8 piano songs composed by *Franz Liszt*) for fine tuning went through the same process as above.

```
import random

def Manually_Change_Velocity(file_name):
    note_items, tempo_items = utils.read_items(file_name+'.midi')
    print(*note_items[:10], sep='\n')

    for note_item in note_items:
        note_item.velocity=random.randint(20,60)
    print(*note_items[:10], sep='\n')
    print(*tempo_items[:10], sep='\n')

    items = tempo_items + note_items
    max_time = note_items[-1].end
    groups = utils.group_items(items, max_time)

    events = utils.item2event(groups)
    midi_path = file_name+'.Velocity_Changed+'.midi
    utils.write_midi_events(None, None, midi_path, prompt_path=None, events=events)
```

```
file_name = "sample_data/[Domenico Scarlatti][Sonata in D Major, K. 118]"
Manually_Change_Velocity(file_name)

Item(name=Note, start=952, end=1044, velocity=127, pitch=74)
Item(name=Note, start=1474, end=1557, velocity=127, pitch=61)
Item(name=Note, start=1935, end=2027, velocity=127, pitch=57)
Item(name=Note, start=2425, end=2518, velocity=127, pitch=62)
Item(name=Note, start=2887, end=2948, velocity=127, pitch=74)
Item(name=Note, start=2887, end=2948, velocity=127, pitch=78)
Item(name=Note, start=3348, end=3440, velocity=127, pitch=73)
Item(name=Note, start=3379, end=3471, velocity=127, pitch=76)
Item(name=Note, start=3502, end=3532, velocity=127, pitch=71)
Item(name=Note, start=3502, end=3563, velocity=127, pitch=74)

Item(name=Note, start=952, end=1044, velocity=32, pitch=74)
Item(name=Note, start=1474, end=1557, velocity=36, pitch=61)
Item(name=Note, start=1935, end=2027, velocity=38, pitch=57)
Item(name=Note, start=2425, end=2518, velocity=55, pitch=62)
Item(name=Note, start=2887, end=2948, velocity=26, pitch=74)
Item(name=Note, start=2887, end=2948, velocity=32, pitch=78)
Item(name=Note, start=3348, end=3440, velocity=32, pitch=73)
Item(name=Note, start=3379, end=3471, velocity=28, pitch=76)
Item(name=Note, start=3502, end=3532, velocity=38, pitch=71)
Item(name=Note, start=3502, end=3563, velocity=45, pitch=74)
```

Figure 1. Process that manually changes velocity value.

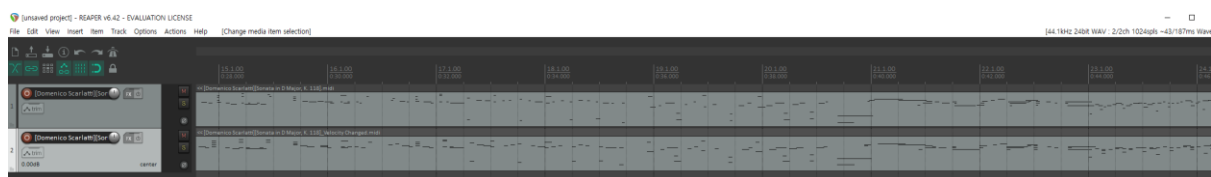


Figure 2. Midi file rendered using Reaper (top: original, bottom: velocity changed).

Next, *Franz Liszt*'s songs transcribed and a part of the Pianist8 dataset (*Bethel*, *Hancock*, *Yiruma*, *Einaudi*, 4 categories) was used for fine-tuning. Rather than collecting all the data and performing fine-tuning at once, each category (total 5=1+4) was independently performed to not affect each other's tuning and to tune to each style. After fine tuning for each category, a piano song *Sonata in D Major, K. 118* was inserted into the model (*Pop Music Transformer*). (See Figure 3).

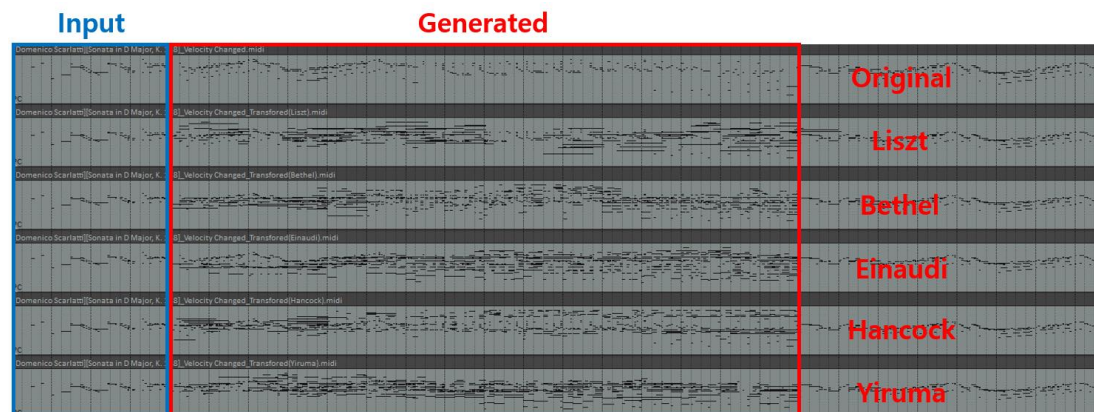


Figure 3. Various generated Midi files (*Sonata in D Major, K. 118*) rendered using Reaper.

Additionally, the experiment was conducted by adjusting the generation parameters such as n_target_bar , $temperature$, and $topk$. n_target_bar means how many bars continuing each prompt the model will create after receiving the first 4 bars as input [1]. For inference, sampling (temperature-controlled stochastic sampling, $p_i = \exp(x_i/T) / \sum_j \exp(x_j/T)$) is performed only on k tokens with the highest (top) probability [2]. More specifically, it can be seen that the bars are randomly generated as $Temperature$ (T) increases. Figure 4 shows results according to various generation parameter setting.

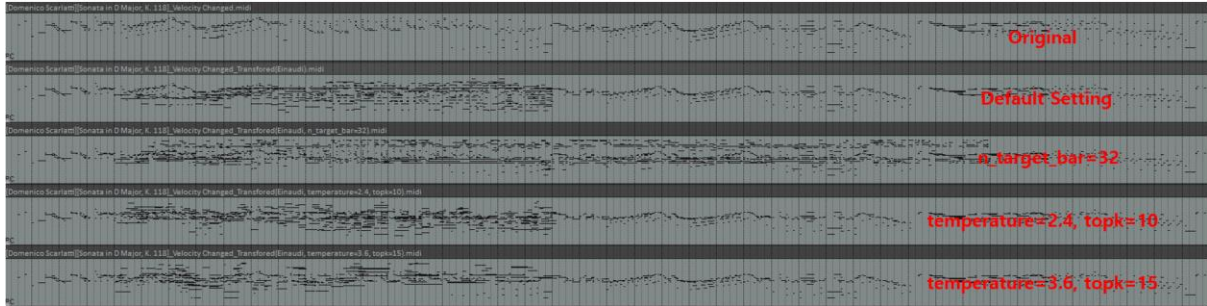


Figure 4. Generated Midi file rendered using Reaper for different generation parameter setting. Default setting is ($n_target_bar=16$, $temperature=1.2$, $topk=5$). Original is only velocity changed.

[Subjective Evaluation]

I directly listened to the generated music and subjectively evaluated it. First of all, as can be seen in Figure 3, the generated results had very rich sound (only in n_target_bar section) in common. However, it was difficult to get a sense of harmony, and I felt that the music suddenly changed at the starting point and ending point of n_target_bar . Personally, I wanted the warm feeling of the songs composed by *Yiruma* to be reflected in the input song, but the result was not and gave me a very mysterious feeling. *Ludovico Einaudi*'s music is reported to be calm, contemplative and concise [3]. However, the results did not meet my personal standards of conciseness and calmness. The cause for these results is speculated that the previously trained songs were too diverse or there were too few songs for fine-tuning. Last thing is about $temperature$ and $topk$. Increasing the value of the these by 3 times from the default setting definitely produced more distracted results.

Reference

- [1] Yu-Siang Huang and Yi-Hsuan Yang. Pop Music Transformer: Beat-based Modeling and Generation of Expressive Pop Piano Compositions. *arXiv*, 2020.
- [2] Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. CTRL: A Conditional Transformer Language Model for Controllable Generation. *arXiv*, 2019.
- [3] https://en.wikipedia.org/wiki/Ludovico_Einaudi