ATTRIBUTE-AWARE DEEP MUSIC TRANSFORMATION FOR POLYPHONIC MUSIC

Yuta Matsuoka
Nagoya Institute of Technology
matsuoka@slp.nitech.ac.jp

Shinji Sako
Nagoya Institute of Technology
s.sako@nitech.ac.jp

ABSTRACT

Recent machine learning technology have made it possible to automatically create a variety of new music. And many approaches have been proposed to control musical attributes such as pitch and rhythm of the generated music. However, most of them focus only on monophonic music. In this study, we apply the deep music transformation model [1], which can control the musical attributes of monophonic music, to polyphonic music. We employ Performance Encoding [2], which can efficiently describe polyphonic music, as the input to the model. To evaluate the proposed method, we performed music transformation using a polyphonic music dataset.

1. INTRODUCTION

Creating music is difficult for many people because it requires music theory and a lot of experience. In such situation, music generation models using machine learning make it possible to create various kinds of music automatically and easily. Therefore, these models can help beginners in music creation. One of them is a model that transforms the input music into the desired music by changing some parameters. This is very useful because it allows you to create music as if you were a composer. However, there are some problems with such music transformation models. One of them is that there are few models that can transform polyphonic music. Since most popular music is polyphonic, it is important to be able to transform polyphonic music. In this study, we employ data in a format that can represent polyphonic music into an existing transformation method that can control musical attributes in monophonic music. Then, we evaluate whether we can achieve the same control as in the case of monophonic music.

2. RELATED WORKS

2.1 Attribute-Aware Music Transformation

Kawai et al. proposed a method to enable attribute-aware music transformation from any set of musical annotations [1]. It uses a model that consists of a Variational Auto Encoder and an adversarial Classifier-Discriminator. The Classifier-Discriminator predicts the music attributes from the latent space. Through adversarial learning, the music attributes in the latent space of the generative model are abstracted. These features are then reintroduced as conditioning to the decoder to control the generation. This method does not require complex derivative implementations and can be used for any form of music attribute. In this paper, we use this method as a music transformation model.

2.2 Performance Encoding

Polyphonic music has no fixed number of notes at the same time, so it contains more information than monophonic music. This makes it difficult to convert polyphonic music into data well. Performance Encoding [2] is a method of serializing the polyphonic musical performance into a sequence of one hot encoded events. This data representation can represent data with high temporal resolution as short sequences, and can be used in various MIDI files. In this paper, we employ this data format because it can deal with polyphonic music without complex implementation.

3. METHODS

3.1 Date Representation

The input MIDI file are encoded into a sequence of events from the following set of vocabulary: 128 NOTE_ON events represent the beginning of a note corresponding to 128 MIDI pitches, 128 NOTE_OFF events to stop the started note, 100 TIME_SHIFT events representing 10ms to 1000ms in 10ms increments, and 32 SET_VELOCITY events with 128 MIDI velocities quantized to 32 bins.

3.2 Music Attributes

Music attributes are musically meaningful values that can be computed from music samples. For example, the total value of notes, the variance of pitches, and so on. The music attributes calculated from each sample are normalized to have zero mean and unit variance. This value is used as the music attribute label $\boldsymbol{a}$. This a is input to the decoder as a conditioning. For training the discriminator, we quantize the music attribute label $\boldsymbol{a}$ to the class label $\boldsymbol{b} \in \{1, 2, ..., K\}$. This K is set to 8 from [1]. This class

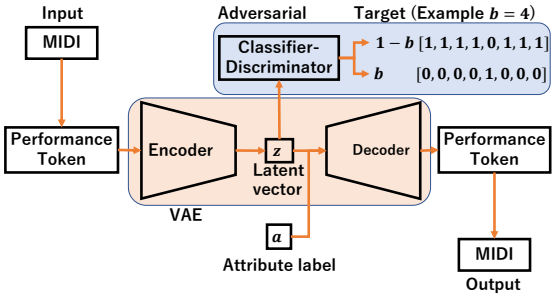

Figure 1. Overview of our model used in this work.

label $\boldsymbol{b}$ is adjusted to make sure that each class contains the same number of data.

3.3 Model Architecture

Figure 1 shows an overview of our model we will use. The input is a sequence of one-hot vectors from the sequence of events described in Section 3.1. The encoder outputs the mean and variance of the latent vactor $\boldsymbol{z}$ from the input event sequences. It consists of a single layer bidirectional Gated Recurrent Unit (GRU) and linear layers that calculates the mean and variance of the latent vector z from GRU output. The decoder reconstructs the output of step t from the output of the previous step $t-1$, the music attributes $\boldsymbol{a}$, and the latent vector $\boldsymbol{z}$. It consists of a two-layer GRU. Classifier-Discriminator predicts to which class label $\boldsymbol{b}$ the latent variable $\boldsymbol{z}$ belongs. It consists of two linear layers, with *tanh* in the first layer and a *sigmoid* function in the second layer as the activation function.

4. EXPERIMENTS

4.1 Dataset

In our experiment, we used 10 years of MIDI data taken from the performance data of Piano-e-Competition [1]. From each music sample, we extracted a segment of four beats at the granularity of a quarter note, with a maximum sequence length of 100. As a result, a total of 171,833 sequences in this dataset are split into training/validation/test sets in a ratio of 80/10/10.

4.2 Baseline

We compare the proposed model in this paper with GLSR-VAE [3] as a baseline. This model removes the Classifier-Discriminator from the proposed model and adds a regularization term to the loss function. The music attribute is controlled by changing one dimension of the latent vector $\boldsymbol{z}$, which is mapped to the attribute by a regularization term. We adopted it as a baseline because [4] has successfully generated polyphonic music by inputting the data representation of Performance Encoding to the model with the mechanism of GLSR-VAE.

[1] Piano-e-Competition dataset (competition history): http://www.piano-e-competition.com/

	GLSR	Ours
accuracy	**0.92**	0.79

Table 1. Reconstruction accuracy

attribute	GLSR	Ours
rhythm density	**0.75**	0.30

Table 2. Spearman's correlation coefficient

4.3 Evaluation

To evaluate the performance of the model, we calculate the reconstruction accuracy and Spearman's ranked correlation coefficient. Spearman's correlation coefficient evaluates whether the change in the input musical attribute label $\boldsymbol{a}$ corresponds linearly to the change in the musical attribute calculated from the output. For the present evaluation, we use rhythm density as the musical attribute to be controlled. It is calculated by dividing the number of onsets in each sequence by the sequence length. Music with a high rhythm density has a sequence of fine notes, while music with a low rhythm density has a wider interval between each note.

4.4 Results

First, we performed the transformation without changing the music attribute labels. The computed reconstruction accuracy is shown in Table 1. The results show that the baseline is better in terms of reconstruction accuracy. We think this is because the baseline controls only one dimension, so there are fewer restrictions on the latent vectors than in the proposed method where all latent vectors are affected by the Classifier-Discriminator. Next, the transformation was performed while changing the input musical attribute labels. Table 2 shows the Spearman's correlation coefficients calculated between the input musical attribute labels and the rhythm densities calculated from the output. The results show that the baseline has more linear control over the music attributes, and the proposed method does not sufficiently separate the music attributes. In this experiment, we used the same setup [1] that was used in the monophonic experiments. Therefore, it may not have been optimized for this polyphonic music setting, and we will continue to experiment with it.

5. CONCLUSION

In this paper, we proposed music transformation model that can control music attributes for polyphonic music. In the future, we would like to optimize not only the input data but also the structure of the model for polyphonic music.

6. REFERENCES

[1] L. Kawai, P. Esling, and T. Harada, "Attributes-aware deep music transformation," in *Ismir 2020 Virtual Conference*, 2020, pp. 670–677.

[2] S. Oore, I. Simon, S. Dieleman, D. Eck, and K. Simonyan, "This time with feeling: learning expressive musical performance," *Neural Computing and Applications*, vol. 32, no. 4, pp. 955–967, 2020.

[3] G. Hadjeres, F. Nielsen, and F. Pachet, "Glsr-vae: Geodesic latent space regularization for variational autoencoder architectures," in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2017, pp. 1–7.

[4] H. H. Tan and D. Herremans, "Music fadernets: Controllable music generation based on high-level features via low-level feature modelling," in *Ismir 2020 Virtual Conference*, 2020, pp. 109–116.