# On the Importance of Time and Pitch Relativity for Transformer-Based Symbolic Music Generation

Tatsuro Inaba*, Kazuyoshi Yoshii*, and Eita Nakamura†
*Kyoto University, Japan
E-mail: inaba@sap.ist.i.kyoto-u.ac.jp, yoshii.kazuyoshi.3r@kyoto-u.ac.jp
†Kyushu University, Japan
E-mail: nakamura@inf.kyushu-u.ac.jp

***Abstract*—This paper describes experimental investigation of music representations to draw the full potential of the Transformer with the self-attention mechanism for symbolic music generation. To use sequence-to-sequence model like the Transformer originally proposed for natural language processing, one typically serializes a musical score into a sequence of event- or note-based tokens without concern for the impact on the quality of generated music. The semantic invariance of music with respect to the time and pitch shifts is attributed to the positional relativity of musical notes over the time-pitch plane in which beats and pitch classes are repeated at intervals of bars and octaves, respectively. We here hypothesize that the capability of the self-attention mechanism to learn the musically meaningful rhythm, melody, and harmony is limited because the relativity and cyclicity of time and pitch information are not explicitly represented in the token sequence. To solve this problem, we propose a cyclicity-aware relative time and pitch encoding unique to music for the attention mechanism. Comprehensive evaluation using the POP909 dataset demonstrated that the proposed Transformer works better with event- or note-based score tokenization[1].**

## I. Introduction

The Transformer [1], a sequence-to-sequence learning model originally proposed for natural language processing (NLP), has revolutionized generative tasks in music information processing [2]. To use the Transformer for symbolic music generation, one needs to represent a musical score as a sequence of *tokens* in an invertible manner because the input and output of the Transformer must be token sequences. In the event-based score tokenization (e.g., MIDI [3] and REMI [4]), each token corresponds to a musical event representing the onset time, instrument, pitch, or duration of a musical note, where multiple events collectively represent a musical note. In the note-based tokenization (e.g., OctapleMIDI [5] and MuMIDI [6]), each token corresponds to a musical note, i.e., a tuple of the onset time, instrument, pitch, and duration of the note. Note that the event-based representation does not guarantee the consistency of token sequences, i.e., token sequences cannot always be inverted to musical scores due to the possibility of missing or overlapping note attributes.

In these representations, the relationships between musical notes over time and pitch are not explicitly considered (Fig. 1). The time relativity is crucial in forming the semantic invariance of music to time shifts. Similarly, the pitch relativity is another key feature of music because a musical piece remains semantically identical with respect to key transposition. The relative index, pitch, and onset (RIPO) attention mechanism [7] is a noticeable approach that considers the time and pitch relativity for sinusoidal encoding of note-based tokens.

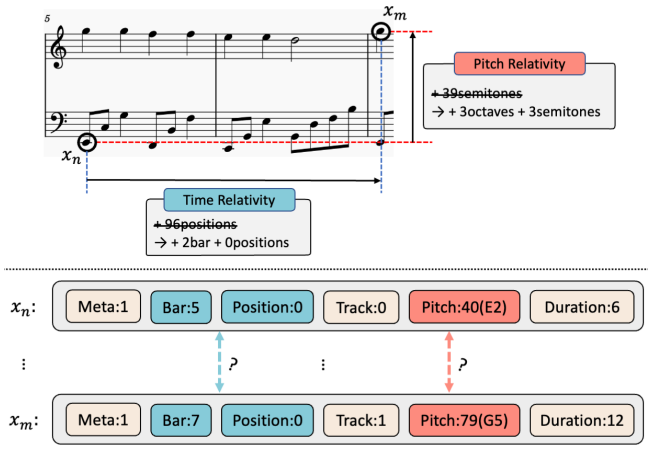


Fig. 1. A token $x_n$ in the note-based representation holds the absolute pitch value of 40 (E2), and another token $x_m$ holds the absolute pitch value of 79 (G5). On the other hand, the relative pitch distance 39 (3 octaves and 3 semitones) is not explicitly captured.

As a piece-agnostic nature of music, beat and pitch circulations play a vital role in music. The bar serves as a fundamental unit of time. Musical themes, for example, tend to be repeated at an interval of 1, 2, 4, or 8 bars. Similarly, the octave serves as a fundamental unit of pitch. Musical sounds whose pitches differ by octaves are perceived as having similar resonance because the harmonic structures of these sounds significantly overlap with each other. As illustrated in Fig. 1, it is musically meaningful to interpret 96 time steps as 2 bars and 39 semitones as 3 octaves plus 3 semitones (a minor 3rd). Appropriate treatment of these fundamental units would thus be considered as a key to achive musically-meaningful music generation compatible with standard music theory.

In this paper, we propose a variant of the attention mechanism named Circular Relative Attention (CirRelAttn) mechanism. Time relativity is decomposed into bar units and their remainders, and pitch relativity into octave units and their remainders. This method can effectively learn the repetitive structures at specific intervals and generates music with en-

[1]The demo page is available at https://tatsuropfgt.github.io/CirRelAttn.

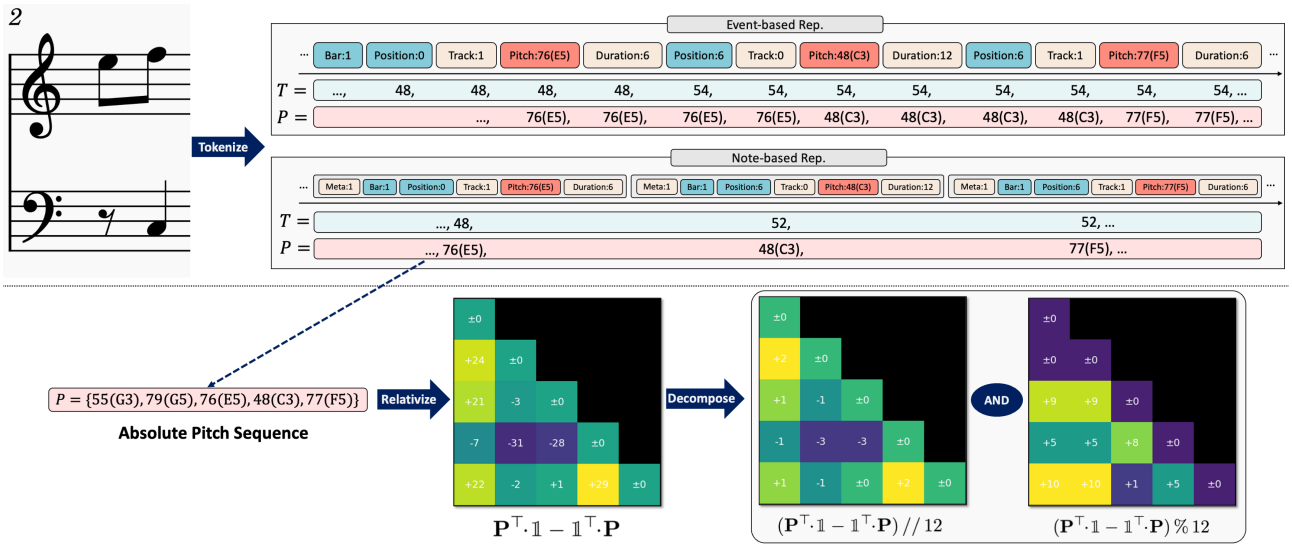


Fig. 2. The top figure shows an example of event-based and note-based representations, along with the absolute time $\mathbf{T}$ and the pitch sequence $\mathbf{P}$. The bottom figure illustrates decomposed relative pitch matrices, which are designed to emphasize the circularity of octaves

hanced coherence. We evaluated our approach using the POP-909 dataset[8], a collection of Chinese popular music rich in repetitive structures. Objective evaluation results showed that our method continually generates musical pieces closer to test data than previous approaches. Additionally, subjective evaluations revealed that human listeners favored the pieces produced by our method.

## II. Related Work

### A. Music Representation

Music representations must be designed carefully to convert a symbolic score of polyhonic music into a one-dimensional sequence of *tokens* that sequence-to-sequence models can directly deal with [2]. This choise is crucial because it has a strong impact on the model ability to learn musical structures. The event-based representations [3], [4] represents musical events such as note-on, note-off (or duration), pitch as tokens. The compound (CP) word representation [9] aggregates musical events based on their specific roles into single tokens. This was extended to the note-based representation [5], [6], where each note is represented by a single token, enhancing the efficiency of sequence length.

### B. Music Structure Modeling

A few approaches have been explored to model the time and pitch structure of music. Chuan and Herremans [10] proposed a method that captures musical relationships between pitches inspired by the Tonnetz method. They encoded these relationships into a 2D representation and utilized a convolutional neural network (CNN) and a long short-term memory (LSTM) network for generating musical scores. Music Transformer [11] claimed the importance of relative information in music and incorporated the index intervals between tokens into the self-attention mechanism. This approach was extended to jointly consider the time and pitch relativity of music with the RIPO attention mechanism [7], which embeds the pitch and onset distances between notes into self-attention mechanism with sinusoidal encoding. Recently, Structure-PE [12] has incorporated the time, section, chord, and melodic pitch relationships between time steps into the self-attention mechanism.

## III. Proposed Method

We define music representations and review the existing self-attention mechanisms (RelAttn and RIPO). We then propose the Circular Relative Attention (CirRelAttn).

### A. Music Representations

We represent a musical score as a sequence of event-based (Section III-A1) or note-based (Section III-A2) tokens denoted by $\mathbf{x} = (\mathbf{x}_1, ..., \mathbf{x}_L)$. For simplicity, we represent a musical score using only essential attributes obtained by omitting performance-related ones (e.g., tempo and velocity) and additional ones (e.g., chord). We also define a time resolution as the 12 steps per quarter note.

*1) Event-based Representation:* We use the events in Table I based on REMI+ [13] representation. The variable $k$ associated with Bar, Position, Track, or Pitch tokens takes an integer within the range. In contrast, $k$ associated with the Duration token takes one of 27 integers specified by the following expression as follows:

$$\begin{aligned} k = &\{1, ..., 12\} \\ &\cup \{12 + 3i \,|\, i \in [1, 4]\} \cup \{12 + 4i \,|\, i \in [1, 3]\} \\ &\cup \{24 + 6i \,|\, i \in [1, 4]\} \cup \{48 + 12i \,|\, i \in [1, 4]\}. \end{aligned} \quad (1)$$

Durations are represented as follows: up to quarter notes for time step lengths; up to half notes for sixteenth notes and triplets; up to whole notes for eighth notes; and up to double whole notes for quarter notes. Unlike REMI+, time-signature, tempo, chord, and velocity events are omitted. In this study, we use a total of $1 + 1 + 16 + 8 + 3 + 128 + 27 = 184$ types of tokens.

TABLE I
TOKEN LIST IN EVENT-BASED REPRESENTATION.

| Event Type | Description | Value Range |
|---|---|---|
| BOS | Beginning of a song | |
| EOS | End of a song | |
| Bar:$k$ | Start of the $k$th bar | $1 \leq k \leq 16$ |
| Position:$k$ | Position of the note in the bar, $k$ | $0 \leq k \leq 47$ |
| Track:$k$ | Track number of the note, $k$ | $1 \leq k \leq 3$ |
| Pitch:$k$ | Pitch of the note, $k$ | $0 \leq k \leq 127$ |
| Duration:$k$ | Duration of the note, $k$ | $1 \leq k \leq 96$ |

The top of Fig. 1 shows an example of the event-based representation. The token sequence begins with a BOS token and ends with an EOS token. Musical notes are sorted in an ascending order of time and pitch to form a series of four tokens: Position, Track, Pitch, and Duration, which are then added to the token sequence in order.

We define the absolute index sequence $\mathbf{I} = \{I_1, ..., I_L\}$, the absolute time sequence $\mathbf{T} = \{T_1, ..., T_L\}$, and the absolute pitch sequence $\mathbf{P} = \{P_1, ..., P_L\}$ in Algorithm 1, which are used to compute the relativity in Sections III-B, III-C, and III-D. *BarRes* represents the timesteps per bar.

*2) Note-based Representation:* This representation is defined as in the Multitrack Music Transformer [6], where each token (note) is expressed as a tuple of 6 variables:

$$\mathbf{x}_i = \{x_i^{\text{meta}}, x_i^{\text{bar}}, x_i^{\text{position}}, x_i^{\text{p}}, x_i^{\text{duration}}, x_i^{\text{track}}\}.$$

Each variable corresponds to Meta, Bar, Position, Track, Pitch, and Duration tokens. Except for Meta and Bar tokens, the definitions are the same as those in the event-based representation (see Section III-A1).

Meta token indicates the beginning of a song (0), a note (1), or the end of a song (2). Bar token represents the bar including a note, with a value range from 1 to 16, consistent with the event-based representation. During encoding, each variable is separately transformed by linear transformation layers, and their sum represents the latent representation of the token. Conversely, during decoding, the latent representation of the output is decoded into each variable separately by six types of linear transformation layers.

The top of Fig. 1 also shows an example of the note-based representation. After converting all musical notes into tokens, tokens are sorted based on time and pitch. A token representing the start of the piece (Meta:0) is added at the beginning, and a token representing the end of the piece (Meta:2) is added at the end of the sorted token sequence. We also define the absolute index, time, and pitch sequences for the note-based representation as follows.

$$\mathbf{I} = \{1, ..., L\}, \tag{2}$$

$$\mathbf{T} = \{x_i^{\text{bar}} \times \textit{BarRes} + x_i^{\text{position}}\}_{i=1}^{L}, \tag{3}$$

$$\mathbf{P} = \{x_i^{\text{pitch}}\}_{i=1}^{L}. \tag{4}$$

### B. Relative Attention

In general, the self-attention layer (Attn) transforms a feature sequence $X \triangleq \{X_1, ..., X_L\} \in \mathbb{R}^{L \times D}$ to another feature sequence $Y \triangleq \{Y_1, \ldots, Y_L\} \in \mathbb{R}^{L \times D}$ while keeping the length $L$ and the dimension $D$. Fist, $X$ is transformed into queries $Q = XW_Q$, keys $K = XW_K$, and values $V = XW_V$, where $W_Q, W_K, W_V \in \mathbb{R}^{D \times D_h}$ are learnable parameters. Here, $D_h$ represents the dimension in an attention head. The original transformer uses the multi-head attention mechanism for capturing the multifaceted relationships between different positions. For simplicity, we here focus on a single attention head. The output of the self-attention layer is given by

$$Y = \text{Softmax}\left(\frac{QK^\top}{\sqrt{D_h}}\right)V \triangleq \text{Attn}(X). \tag{5}$$

On the other hand, relative attention [11], [14] incorporates positional embeddings derived from relative index distances between tokens into the attention mechanism.

$$\mathbf{R}_{\text{idx}} = \text{LPE}(\mathbf{I}^\top \cdot \mathbb{1} - \mathbb{1}^\top \cdot \mathbf{I}) \in \mathbb{R}^{L \times L \times D_h}, \tag{6}$$

$$S_{\text{rel}}^{\text{idx}} = Q\mathbf{R}_{\text{idx}}^\top \in \mathbb{R}^{L \times L}, \tag{7}$$

$$Y = \text{Softmax}\left(\frac{QK^\top + \alpha S_{\text{rel}}^{\text{idx}}}{\sqrt{D_h}}\right)V \triangleq \text{RelAttn}(X), \tag{8}$$

where $\mathbb{1}$ is an all-one vector of length $L$ and $\alpha$ is a hyperparameter used for scaling. The relative index matrix $\mathbf{I}^\top \cdot \mathbb{1} - \mathbb{1}^\top \cdot \mathbf{I}$ is embedded using learnable positional encoding LPE. Subsequently, the relative index logit $S_{\text{rel}}^{\text{idx}}$, computed by the product of the embedding $\mathbf{R}_{\text{idx}}$ and the query $Q$, is scaled by $\alpha$ and added to $QK^\top$. We refer to relative attention as RelAttn.

### C. RIPO Attention

RIPO attention incorporates the relative time and pitch distances between notes in original music as well as the relative index distances between tokens.

$$\mathbf{R}_{\text{t}} = \text{SPE}(\mathbf{T}^\top \cdot \mathbb{1} - \mathbb{1}^\top \cdot \mathbf{T}) \in \mathbb{R}^{L \times L \times D_h}, \tag{9}$$

$$\mathbf{R}_{\text{p}} = \text{SPE}(\mathbf{P}^\top \cdot \mathbb{1} - \mathbb{1}^\top \cdot \mathbf{P}) \in \mathbb{R}^{L \times L \times D_h}, \tag{10}$$

**Algorithm 1** Event-based $\mathbf{I}, \mathbf{T}, \mathbf{P}$

```
cbar ← None #current bar
cpos ← None #current position
cpit ← None #current pitch
I, T, P ← [], [], []
for idx, event ∈ enumerate(sequence) do
    if type of event is Bar then
        cbar ← Bar value of event
    end if
    if type of event is Position then
        cpos ← Position value of event
    end if
    if type of event is Pitch then
        cpit ← Pitch value of event
    end if
    I.append(idx)
    T.append(cbar × BarRes + cpos)
    P.append(cpit)
end for
```

$$S^{\mathrm{t}}_{\mathrm{rel}} = Q\mathbf{R}^{\top}_{\mathrm{t}} \in \mathbb{R}^{L\times L}, \tag{11}$$

$$S^{\mathrm{p}}_{\mathrm{rel}} = Q\mathbf{R}^{\top}_{\mathrm{p}} \in \mathbb{R}^{L\times L}, \tag{12}$$

$$S_{\mathrm{rel}} = S^{\mathrm{idx}}_{\mathrm{rel}} + S^{\mathrm{t}}_{\mathrm{rel}} + S^{\mathrm{p}}_{\mathrm{rel}} \in \mathbb{R}^{L\times L}, \tag{13}$$

$$Y = \mathrm{Softmax}\left(\frac{QK^{\top} + \alpha S_{\mathrm{rel}}}{\sqrt{D_h}}\right)V \triangleq \mathrm{RIPOAttn}(\mathbf{X}). \tag{14}$$

The relative time matrix $\mathbf{T}^{\top}\cdot\mathbb{1} - \mathbb{1}^{\top}\cdot\mathbf{T}$ and the relative pitch matrix $\mathbf{P}^{\top}\cdot\mathbb{1} - \mathbb{1}^{\top}\cdot\mathbf{P}$ are embedded with the sinusoidal positional encoding SPE. The embeddings $\mathbf{R}_{\mathrm{t}}$ and $\mathbf{R}_{\mathrm{p}}$ are then utilized to compute the relative time logit $S^{\mathrm{t}}_{\mathrm{rel}}$ and the relative pitch logit $S^{\mathrm{p}}_{\mathrm{rel}}$ by performing the matrix product with the query $Q$. These logits are subsequently scaled by $\alpha$ and added to $QK^{\top}$.

### D. Circular Relative Attention

Leveraging the inherent cyclicity in time and pitch, the proposed CirRelAttn mechanism effectively models the repetitive and cyclic structures that weren't captured by RIPO attention. Initially, we decompose the relative time distance matrix into bar units and their remainders, and then separately embed these matrices using learnable positional encodings.

$$\mathbf{R}_{\mathrm{bar}} = \mathrm{LPE}((\mathbf{T}^{\top}\cdot\mathbb{1} - \mathbb{1}^{\top}\cdot\mathbf{T}) \,//\, \textit{BarRes}), \tag{15}$$

$$\mathbf{R}_{\mathrm{position}} = \mathrm{LPE}((\mathbf{T}^{\top}\cdot\mathbb{1} - \mathbb{1}^{\top}\cdot\mathbf{T}) \,\%\, \textit{BarRes}). \tag{16}$$

Similarly, we decompose relative pitch distance into octave units and their reminders and separately embed these matrices using learnable positional encoding.

$$\mathbf{R}_{\mathrm{octave}} = \mathrm{LPE}((\mathbf{P}^{\top}\cdot\mathbb{1} - \mathbb{1}^{\top}\cdot\mathbf{P}) \,//\, 12), \tag{17}$$

$$\mathbf{R}_{\mathrm{semitone}} = \mathrm{LPE}(\mathbf{P}^{\top}\cdot\mathbb{1} - \mathbb{1}^{\top}\cdot\mathbf{P}) \,\%\, 12). \tag{18}$$

Here, the reason for not using sinusoidal positional encoding is that proximity in relative distance does not necessarily imply a strong relationship. For example, as pointed out in [15], similar musical structures tend to appear in specific previous bars (1, 2, 4, 8, 16, ... bars). Moreover, the harmonic relationship between the pitches is intricate, resulting in each interval having a unique resonance. We utilize learnable positional encoding LPE to capture the features corresponding to these relative distances separately. The relative time and pitch logits from the aforementioned embeddings are represented by one of the following equations.

$$\begin{cases} S^{\mathrm{t}}_{\mathrm{rel}} = Q\left(\mathbf{R}_{\mathrm{bar}} + \mathbf{R}_{\mathrm{position}}\right)^{\top}, \\ S^{\mathrm{p}}_{\mathrm{rel}} = Q\left(\mathbf{R}_{\mathrm{octave}} + \mathbf{R}_{\mathrm{semitone}}\right)^{\top}, \end{cases} \tag{19}$$

$$\begin{cases} S^{\mathrm{t}}_{\mathrm{rel}} = Q\left(\mathbf{R}_{\mathrm{bar}} \odot \mathbf{R}_{\mathrm{position}}\right)^{\top}, \\ S^{\mathrm{p}}_{\mathrm{rel}} = Q\left(\mathbf{R}_{\mathrm{octave}} \odot \mathbf{R}_{\mathrm{semitone}}\right)^{\top}. \end{cases} \tag{20}$$

Here, we define equation (19) as CirRelAttn-S and equation (20) as CirRelAttn-H.

## IV. Evaluation

In this section, we describe the objective and subjective evaluation results from the continuous generation experiments and analyze the effectiveness of the proposed method.

### A. Experimental Setup

We used POP909 dataset [8], which contains popular music with a wide variety of repetitive structures. We excluded songs with inconsistent beat annotations and split the available 896 songs into 10% for validation, 10% for testing, and the remaining 80% for training. We extracted sequences of 16 measures from each song with a stride of 1 measure and used only sequences with 4/4 time signature. Finally, we obtained 4,749 sequences for validation, 4,137 for testing, and 35,452 for training. We quantized the data with a resolution of 12 steps per quarter note. We used MusPy [16] for data pre-processing. For training, we performed data augmentation by randomly transposing each data between -6 to +5 semitones.

For the proposed method, we used a decoder-only Transformer based on a self-attention mechanism with 4 layers, 8 heads, a hidden dimension of 256, and a dropout rate of 0.2. For comparison, we tested the vanilla attention mechanism (Attn) [1], the relative attention mechanism (RelAttn) [11], and the RIPO attention mechanism (RIPOAttn) [7], where the same architecture was used as the proposed method. We conducted validation every 1,000 steps and terminated training after 200,000 steps or if no improvement in loss was observed after 20 validation checks. We fixed the batch size at 8, with a warm-up step count of 10,000 and a peak learning rate of 2e-5. We set the hyperparameter to $\alpha = 0.1$.

### B. Objective Evaluation

We evaluated how accurately the model can continuously generate musical pieces that resemble the test data. The model was given the first 15 bars of each test dataset and generated a continuation for 1 bar. We calculated the similarity between the generated musical pieces and the ground truth using the following five objective metrics.

- NoteF1 ($F1_{\mathrm{note}}$) or OnsetF1 [17] measures the accuracy of the generated pieces against the ground truth on a per-note basis using the F1 score. A true positive is identified when the onset timing, pitch, and track of the generated note all match the note in the ground truth.
- PianorollF1 ($F1_{pr}$) or FrameF1 [17] measures the accuracy of the generated pieces against the ground truth on the piano roll representation by computing F1 score by evaluating each time step and pitch class.
- Grooving Similarity (GS) [18], [19] is the cosine similarity between the grooving vectors of the generated music and the ground truth. Each grooving vector represents the number of onsets at each time step within a measure and is a 48-dimensional vector in this experiment.
- Chroma Similarity (CS) [18] is the cosine similarity between the chroma vectors [20] of the generated pieces and the ground truth. The chroma vector [20] is a 12-dimensional vector representing the number of onsets for each of the 12 pitch classes (C, C#, ..., B) within a specific time range. In this experiment, the cosine similarity is calculated for each half-bar (24 time steps) and the average is computed.

TABLE II
OBJECTIVE EVALUATION RESULTS.

| No. | Rep. | Methods: Attn Type | params | time (ms/note) | Objective Evaluation: Loss | $F1_{note}$ | $F1_{pr}$ | GS | CS | PRS |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | Attn [1] | 4.32M | 8.46 | 0.979 | 0.174 | 0.239 | 0.846 | 0.620 | 0.955 |
| 2 | | RelAttn [11] | 4.84M | 9.73 | 0.937 | 0.218 | 0.291 | 0.848 | 0.650 | 0.955 |
| 3 | event | RIPOAttn [7] | 4.85M | 18.3 | 0.904 | 0.233 | 0.305 | 0.855 | 0.660 | 0.956 |
| 4 | | CirRelAttn-S | 4.88M | 22.5 | 0.876 | 0.268 | 0.341 | 0.855 | 0.673 | 0.957 |
| 5 | | CirRelAttn-H | 4.88M | 20.8 | **0.856** | **0.293** | **0.361** | **0.858** | **0.685** | **0.958** |
| 6 | | Attn [1] | 3.53M | 4.29 | 4.42 | 0.188 | 0.267 | 0.784 | 0.619 | 0.941 |
| 7 | | RelAttn [11] | 3.66M | 5.79 | 4.38 | 0.186 | 0.271 | **0.798** | 0.628 | **0.946** |
| 8 | note | RIPOAttn [7] | 3.67M | 5.93 | 4.40 | 0.180 | 0.257 | 0.786 | 0.612 | 0.942 |
| 9 | | CirRelAttn-S | 3.70M | 8.58 | 4.37 | 0.192 | 0.273 | 0.785 | 0.623 | 0.943 |
| 10 | | CirRelAttn-H | 3.70M | 6.54 | **4.29** | **0.215** | **0.294** | 0.787 | **0.632** | 0.944 |

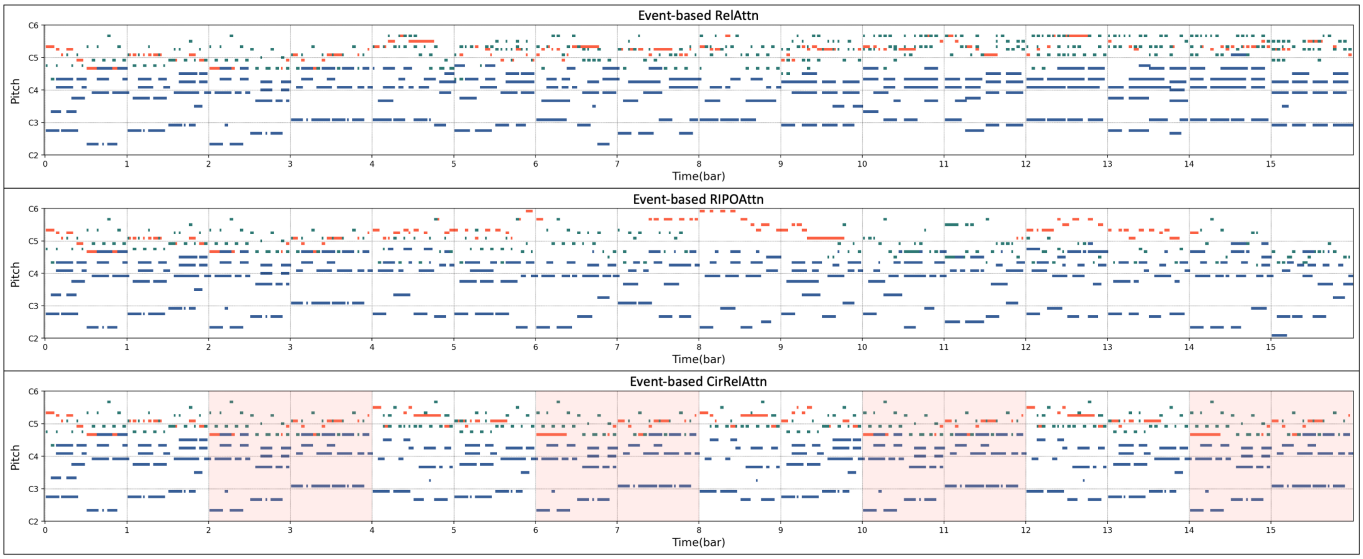


Fig. 3. Musical pieces generated by the event-based RelAttn, RIPOAttn, and CirRelAttn-H. Given the first four bars, each model generates a continuation for the next twelve bars. While the RelAttn and RIPOAttn generate entirely new phrases, CirRelAttn generates phrases that resemble the initial input, resulting in more coherent music. The areas shaded in red indicate a repetitive structure.

TABLE III
SUBJECTIVE EVALUATION RESULTS.

| Methods: Rep. | Attn Type | Subjective Evaluation: Coherence | Musicality | Overall |
|---|---|---|---|---|
| Ground Truth | | 3.93 | 4.17 | 4.03 |
| event | RelAttn [11] | 2.93 | 2.69 | 2.79 |
| | RIPOAttn [7] | 3.00 | 3.21 | 3.03 |
| | CirRelAttn-H | **4.31** | **3.41** | **3.69** |
| note | RelAttn [11] | 2.28 | 2.69 | 2.45 |
| | RIPOAttn [7] | **2.69** | **2.90** | **2.90** |
| | CirRelAttn-H | 2.10 | 2.51 | 2.42 |

- Pitch Range Similarity (PRS) measures the similarity of the pitch range within a single bar between the generated pieces and the ground truth.

$$\text{PRS} = 1 - \frac{|\text{PR}_{\text{gen}} - \text{PR}_{\text{gt}}|}{128}, \tag{21}$$

where $\text{PR}_{\text{gen}}$ and $\text{PR}_{\text{gt}}$ represent the pitch range (the difference between the highest and lowest pitches) of the generated pieces and the ground truth, respectively.

Table II shows the objective evaluation results on the continuous generation task. Our methods (No. 5, 10) demonstrated superior performance to the baselines on most objective metrics within both event-based and note-based representations. In other words, we confirmed that the proposed methods generate musical pieces that more closely resemble the test data compared to the baselines.

### C. Subjective Evaluation

We conducted a listening test to evaluate the quality of musical pieces generated by our method. We compared the generated music pieces by six models (event-based RelAttn, RIPOAttn, and CirRelAttn and note-based RelAttn, RIPOAttn, and CirRelAttn) and the ground truth. We provided only the initial four bars to each model, and the model then generated the continuation for twelve bars. Each evaluator listened to seven pieces and scored them based on three criteria: coherence, musicality, and overall quality, using a scale from 1 to 5. We recruited 30 evaluators to evaluate 10 sets of musical pieces, with each set scored by three evaluators.

Table III shows the result of the listening test. Our method with the event-based representation (CirRelAttn-H) outperformed other methods, particularly in coherence. We found

that CirRelAttn-H with the event-based representation tended to generate repetitive structures by analyzing the generated musical pieces. Fig. 3 shows an example of musical pieces generated by RelAttn, RIPOAttn, and CirRelAttn. While RelAttn and RIPOAttn constantly generated new phrases, CirRelAttn repeatedly generated the phrase given in the first four bars. We hypothesized that music containing a high amount of repetitive structures received particularly high ratings for coherence from human evaluators. The proposed attention mechanism enabled the model to learn the distinctive repetitive structure of music well and generate consistent music.

## V. Conclusion

This paper presented a novel approach that decomposes the relative time and pitch distances based on bar and octave units and subsequently integrates these components into the self-attention mechanism via learnable positional encodings. Objective evaluation demonstrated that our method effectively learns music structures and outperforms previous methods in a music continuation task. We also confirmed that our method generates coherent music that contains repetitive structures through a listening test.

## Acknowledgment

This work was partially supported by JST FOREST Nos. JPMJFR2270 and JPMJFR226X, JST PRESTO JPMJPR20CB, and JSPS KAKENHI Nos. 24H00742, 24H00748, and 22H03661.

## References

[1] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," in *Neural Information Processing Systems*, 2017, pp. 5998–6008.

[2] S. Ji, X. Yang, and J. Luo, "A survey on deep learning for symbolic music generation: Representations, algorithms, evaluations, and challenges," *ACM Comput. Surv.*, vol. 56, no. 1, 2023.

[3] 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, pp. 955–967, 2018.

[4] Y.-S. Huang and Y.-H. Yang, "Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions," in *ACM International Conference on Multimedia*, 2020, pp. 1180–1188.

[5] M. Zeng, X. Tan, R. Wang, Z. Ju, T. Qin, and T.-Y. Liu, "MusicBERT: Symbolic music understanding with large-scale pre-training," in *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 2021, pp. 791–800.

[6] H.-W. Dong, K. Chen, S. Dubnov, J. McAuley, and T. Berg-Kirkpatrick, "Multitrack music transformer," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[7] Z. Guo, J. Kang, and D. Herremans, "A domain-knowledge-inspired music embedding space and a novel attention mechanism for symbolic music modeling," in *AAAI Conference on Artificial Intelligence*, 2023.

[8] Z. Wang, K. Chen, J. Jiang, *et al.*, "Pop909: A pop-song dataset for music arrangement generation," in *International Society for Music Information Retrieval Conference ISMIR*, 2020.

[9] W.-Y. Hsiao, J.-Y. Liu, Y.-C. Yeh, and Y.-H. Yang, "Compound word transformer: Learning to compose full-song music over dynamic directed hypergraphs," in *AAAI Conference on Artificial Intelligence*, 2021.

[10] C.-H. Chuan and D. Herremans, "Modeling temporal tonal relations in polyphonic music through deep networks with a novel image-based representation," *AAAI Conference on Artificial Intelligence*, 2018.

[11] C.-Z. A. Huang, A. Vaswani, J. Uszkoreit, *et al.*, "Music transformer," in *International Conference on Learning Representations ICLR*, 2019.

[12] M. Agarwal, C. Wang, and G. Richard, "Structure-informed positional encoding for music generation," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024.

[13] D. von Rütte, L. Biggio, Y. Kilcher, and T. Hofmann, "FIGARO: Controllable music generation using learned and expert features," in *International Conference on Learning Representations ICLR*, 2023.

[14] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," in *North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 2018, pp. 464–468.

[15] B. Yu, P. Lu, R. Wang, *et al.*, "Museformer: Transformer with fine- and coarse-grained attention for music generation," in *Neural Information Processing Systems*, 2022.

[16] H.-W. Dong, K. Chen, J. McAuley, and T. Berg-Kirkpatrick, "Muspy: A toolkit for symbolic music generation," in *International Society for Music Information Retrieval Conference ISMIR*, 2020.

[17] J. P. Gardner, I. Simon, E. Manilow, C. Hawthorne, and J. Engel, "MT3: Multi-task multitrack music transcription," in *International Conference on Learning Representations ICLR*, 2022.

[18] S.-L. Wu and Y.-H. Yang, "MuseMorphose: Full-song and fine-grained piano music style transfer with one Transformer VAE," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.

[19] S. Dixon, F. Gouyon, and G. Widmer, "Towards characterisation of music via rhythmic patterns," in *International Society for Music Information Retrieval Conference ISMIR*, 2004.

[20] T. Fujishima, "Realtime chord recognition of musical sound : A system using common lisp music," *International Computer Music Conference ICMC*, pp. 464–467, 1999.