國家科學及技術委員會補助

大專學生研究計畫研究成果報告

計　畫
名　稱：使用 Video Diffusion Model 生成鋼琴流行音樂

報 告 類 別 ：成果報告
執行計畫學生：程品奕
學生計畫編號：NSTC 112-2813-C-006-017-E
研 究 期 間 ：112年07月01日至113年02月29日止，計8個月
指 導 教 授 ：蘇文鈺

處 理 方 式 ：本計畫可公開查詢

執 行 單 位 ：國立成功大學資訊工程學系（所）

中 華 民 國　113年03月14日

Table of Contents

Abstract

While numerous variants of Transformers and image diffusion models have been proposed for piano music generation, there is always a gap between the generated music and the real music in terms of structuredness. In this research, we propose a generative model for pop piano music based on the Video Diffusion Model. Different from the image-like representations that treat a whole piano roll as an image, our video-like representation decomposes the dimensions of sub-beats and bars, emphasizing both music's local structures inside a bar and patterns across bars. The temporal attention blocks we add to our model make the model aware of patterns across bars, meanwhile greatly reducing the window size of the spatial attention compared to the image diffusion baseline (reduce from 512 to 32). Based on the evaluations, our model outperforms both the CP Transformer and the image diffusion baseline in terms of introducing richer patterns across bars and generating music that is preferred by humans.

目前已有許多 Transformer 和 image diffusion model 的變體被提出作為鋼琴音樂生成模型，但它們生成的音樂與真實音樂在結構性上仍然存在著差距。在本研究中，我們提出了一種基於 Video Diffusion Model 的鋼琴流行音樂生成模型。有別於 image-like representation 將 piano roll 視為一整個圖像，我們使用的 video-like representation 將節拍和小節獨立成兩個維度，分別凸顯出小節內的局部結構和橫跨多個小節的模式。我們在模型中添加的 temporal attention blocks 使模型能感知到橫跨多個小節的模式，而且與 image diffusion baseline 相比，這樣做大大減少了 spatial attention 的窗口大小（從 512 減少到 32）。根據實驗結果，我們的模型比 CP Transformer 和 image diffusion baseline 更擅長生成橫跨小節的模式，且其生成的音樂較被聽者所偏好。

Keywords: deep learning, diffusion model, music generation

關鍵詞: 深度學習、擴散模型、音樂生成

1 Preface

In the field of symbolic music generation, there are two mainstream ways to represent music data: text-like representations and image-like representations. Most SOTA approaches [1][2] convert music into MIDI-like sequences, which can be thought of as text written in some "music language", and train Transformers or RNNs to generate them auto-regressively. Others [3][4] see music as piano rolls, which can be thought of as images, and train diffusion models to generate them.

Both text-like and image-like representations have their downsides. Text-like approaches require tokens to be generated in pre-defined orders, requiring additional tricks to enable various tasks such as infilling or accompaniment. Also, since the text-like representation flattens notes and metrics into 1-D sequences, mixing them together, local structures such as harmonies and rhythms, which are important to human perception, cannot be represented explicitly. Image-like approaches solve the above-mentioned problem, but surfer poor long-term structuredness.

To get over these downsides, we propose a novel representation that treats music as videos (Figure 1.). In our proposed method, we convert each 16-bar piano music piece into 16 piano rolls, stack them together, and quantize each bar into 32 timesteps to get a tensor with dimension of (16, 32, 88). As such, it can be seen as a video with 16 frames, 32 pixels, and 88 channels. Such representation combines the idea of image and sequence, explicitly presents both local structure and inter-bar connection of music and enables the model to directly learn them. With this data, we train a variant of Video Diffusion Model [5]. Video Diffusion Model is a class of Denoising Diffusion Probabilistic Model (DDPM) that is specialized to generate videos with both good image quality and temporal coherence with its spatial and temporal self-attention mechanism.

2 Research Objectives

The objective is to construct a model that can generate 16-bar-long piano pop music with good quality, where the measure of quality is defined in the section **4.4 Evaluation**. The model we propose (hereafter referred to as “our model”) is a VDM-like DDPM that utilizes temporal attention. We also construct a baseline image diffusion model that is close to the models used in the image-like approaches. The baseline model’s architecture is identical to ours except that it lacks the temporal attention blocks.

3 Literature Review

3.1 Transformer-based Piano Music Generative Models

Transformers have been a prominent choice for symbolic music generation for years. Most of the works that generate symbolic piano music with Transformer design their data representation based on MIDI event sequence. REMI [1] adopts a representation that is an improved version of MIDI events closer to how humans read music. In their approach, a played note is represented as

a flat sequence of 4 tokens: Position, Note Velocity, Note On, and Note Duration. Another paper, Compound Word Transformer (CP) [2], points out that different types of events that are used in REMI should be treated differently. Instead of flattening all events into a sequence, they group up related events into two types of tokens, note-related and metric-related. Each event in a token is processed by the Transformer model simultaneously but with different feed-forward head. The improved representation explicitly presents the types of events and greatly reduces the sequence length of the data, thus improving the performance of generation.

3.2 Diffusion-based Piano Music Generative Models

Recently, as diffusion models are showing great success in high-quality image generation, some research tries to develop diffusion model's potential in the domain of music generation. Although diffusion models are originally designed to work on continuous domain, [6] finds a way to generate symbolic music with them by parameterizing the discrete domain (MIDI event sequence) into a continuous latent space with MusicVAE, a variational autoencoder that is trained on symbolic music. Two most recent works further show that diffusion models can work on piano rolls, that is, the models can directly generate music notes, no need for an intermediate latent space. DiffRoll [3] treats a piano roll as a tensor of dimension (88, τ) as a 1-D sequence with 88 channels and length τ, where τ is the number of time frames in the piano roll. The data is then used to train a gaussian diffusion model based on 1-D convolutions. SDMuse [4] simply treats a piano roll as a 2-D image and trains a stochastic differential equation (SDE) diffusion model with U-Net architecture (Çiçek et al.,2016) to generate them. Additionally, they trained a separate Transformer decoder to refine the output of the diffusion model to get the final MIDI data with more precise details. According to the paper, SDMuse achieves SOTA generation quality, its objective and subjective metrics are both slightly better than the Transformer-based models REMI and CP.

Because SDMuse and DiffRoll treat music data as 1-D or 2-D images and model it with image generative model, we refer this kind of representation as "image-like representation".

An important strength of diffusion models, as SDMuse demonstrates, is that they can do unconditional generation and various conditional generation tasks, such as continuation and inpainting, without additional training. Generally speaking, diffusion-based models that have been trained on piano rolls can reconstruct piano rolls that are masked with arbitrary shapes. In contrast, auto-regressive Transformer-based models, such as REMI and CPW are limited to generating in a pre-defined order and require additional modification and re-training to perform conditional generation tasks other than continuation.

3.3 Video Diffusion Models

Video Diffusion Model (VDM) [5] is a class of Denoising Diffusion Probabilistic Model (DDPM) specialized to generate videos with both good image quality and temporal coherence. The idea of VDM is to extend the existing image diffusion model on the time dimension. A typical network architecture for 2-D image diffusion model is U-Net with 2-D convolution and spatial self-attention. To3. tackle the extra time dimension in videos, VDM adds 1-D temporal

self-attention blocks that operate on the time dimension, while keeping the 2-D spatial convolution and self-attention blocks. The temporal attention block connects each pixel at the same position across frames, so VDM can learn to generate video that is coherent in all local areas, e.g., no object disappears abruptly.

We found that the idea of VDM is excellent for generating piano roll music. In our proposed video-like representation, we see each bar in a piano roll as a frame in a video. Also, our proposed model adopts both spatial and temporal self-attention blocks, just like VDM. As such, the timesteps on the same beat (or sub-beat) across different bars is directly connected with the temporal attention block. We expect this setting will make the model better learn repetition patterns across bars.

3.4 Comparison of Different Representations

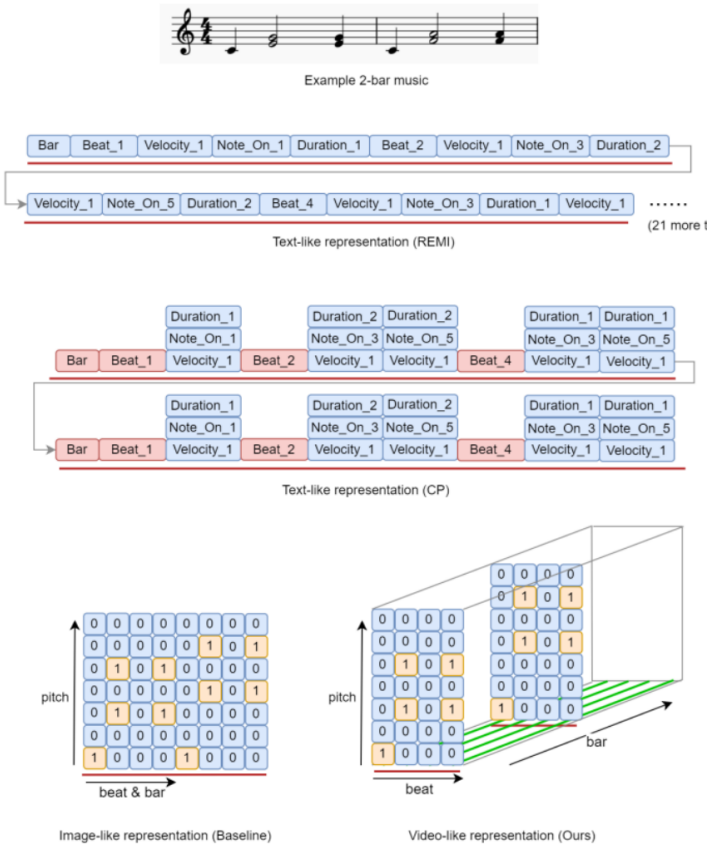

Figure 1. Representing an example 2-bar music using text-like representation, image-like representation, and video-like representation respectively.

Figure 1. shows how the representations proposed in previous works and the video-like representation we propose differently express a piece of 2-bar music. All REMI, CP, SDMuse and DiffRoll use a single set of self-attention or convolution blocks to model the whole sequence of music. In contrast, our approach, adopting video-like representation, decomposes beats and bars into two separate dimensions and applies two separate sets of self-attention blocks to handle the two dimensions (the green lines and the red lines in Figure 1.). By doing that, we make the

sequence length that the self-attention blocks must take care of dramatically reduces. Also, the temporal attention (the green lines in Figure 1.) directly connects the entries in different bars but on the same beat, encouraging repeating rhythms across bars to be learned. With these two benefits on our representation and model, we believe that our approach can achieve much better sample quality compared to previous approaches.

4 Methodology

4.1 Dataset

We collect audio files of 2444 pieces of pop music piano covers from YouTube. We convert each audio file into the piano roll representation used in this research through the following pipeline (Figure 2.):

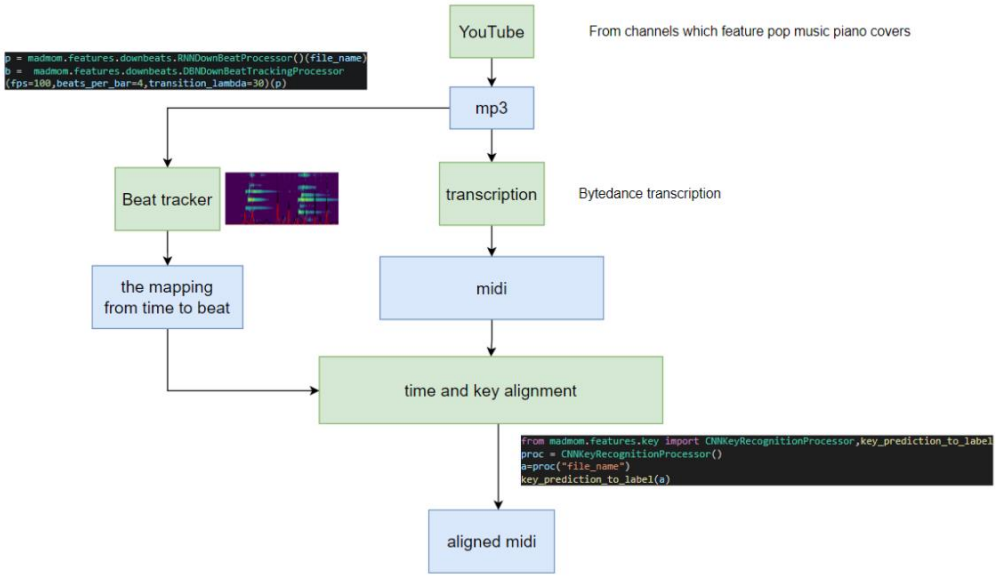

Figure 2. The data collection pipeline used to download and process music data for training and testing purposes.

(1) Transcription: The audio data is first transcribed into midi with a SOTA piano music transcription algorithm [7].

(2) Beat Tracking: We used Madmom, a python library for music analysis, to extract the timestamps of each beat and downbeat events in the audio data.

(3) Time Alignment: The mapping from beats to timestamps we get from the last step is then reversed, linearly interpolated, and quantized to get another mapping that maps time to its corresponding sub-beat index. Using this mapping, we convert the onset timestamps in the midi data into sub-beat indices. The granularity of quantization is 1/8 beat. Because all song in our dataset has 4 beats per bar, there are 32 sub-beats indices pre bar.

(4) Key Alignment: Different songs may have different key signatures from the data source. However, to our knowledge, the key signature of a piece of music has little importance to human perception and is nearly independent from other variables of the piece. Therefore, we marginalize out the key difference by shifting all pieces to C major or A minor. This enables our model to learn scale degrees and octaves instead of absolute pitches, which is closer to how humans naturally understand music.

Finally, we get the dataset of 2444 pieces in the piano roll representation. Each piano roll is a tensor with shape (n_bar, 32, 88), where n_bar is the number of bars. The first, second and third dimensions represent bar, sub-beat, and pitch, respectively. For each entry in the piano roll, if there exists an onset event occurs at the corresponding bar, sub-beat, and pitch, its value will be the velocity of that onset. Otherwise, the entry will be zero.

4.2 Model Architecture

We use the opensource codebase of Improved Denoising Diffusion Probabilistic Models [8] as the DDPM framework.

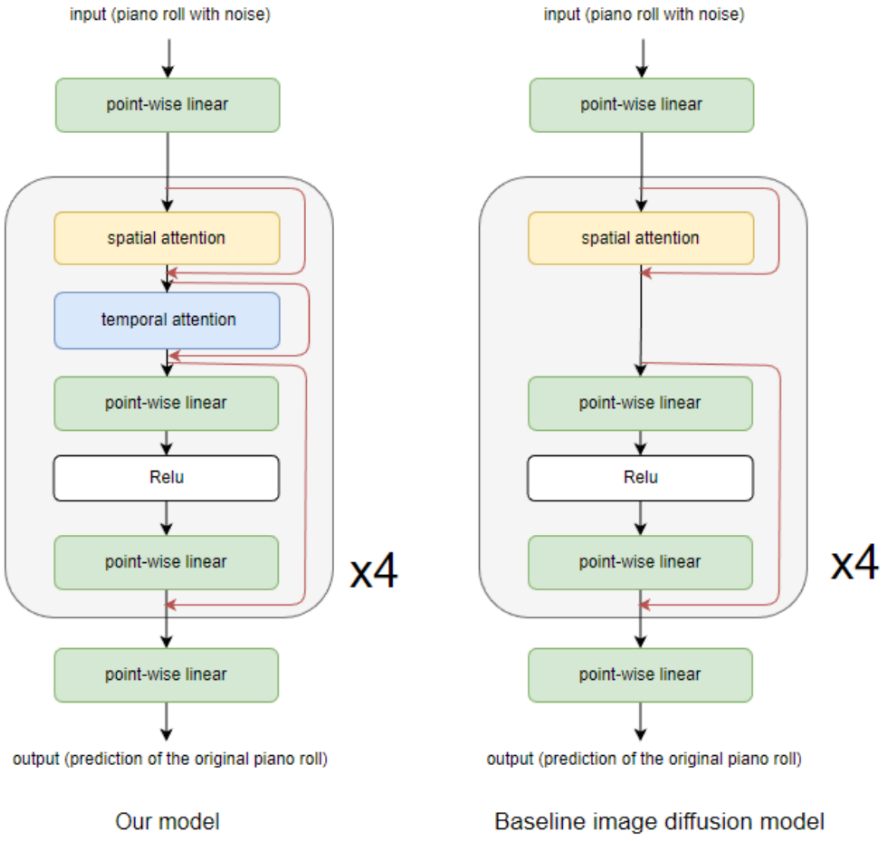

Figure 3. Comparison between the neural network architectures of our model (video diffusion model) and the baseline image diffusion model.

For our model, the model architecture is a 4-layer transformer encoder. Each layer is a sequential composition of a bar-wise spatial self-attention layer, a temporal self-attention layer that connects across bars, and two point-wise feed-forward layers. The input and output layers have 88 features, and all the hidden layers have 512 features. We use self-attention instead of CNN and U-Net to handle sub-beat dimension because at the granularity of 1/8 beat, relevant features usually appear numerous units apart across the sub-beat axis. Also, compared to the attention window size in REMI (512), our window size for attention is only 32, which is computational efficient.

The spatial attention blocks in our model are designed to process each bar separately, that is, the attention window size is 32. However, the baseline image diffusion model lacks temporal attention, so the spatial attention blocks in it must handle all 16 bars together, making the attention window size be 32 * 16 = 512.

4.3 Training

We train both our model and the baseline image diffusion model for 96 hours on 8 V100s before evaluating them. We find both models' performance (both loss and sample quality) keeps improving in the 96 hours, but we have to early stop due to limited computation resources.

In VDM's approach, although the model's purpose is to generate videos, the model can first be trained to generate images for lots of iterations before being trained to generate videos. In our case, we thought that this trick could reduce the training time. We thought the expected autocorrelation of a batch of 16 random 1-bar piano rolls must be lower than that of 1 random 16-bar piano roll drawn from a continuous slice of a music piece. Thus, training the model with the former yields stabler gradient, which is beneficial for the model to converge. However, after an experiment, we find there's no significant difference between the two training schemes. Therefore, we only pick the model that is completely trained with 16-bar piano rolls for evaluation.

4.4 Evaluation

In the objective evaluation and the subjective evaluation, three models are compared: our model, image diffusion baseline, and CP.

4.4.1 Objective evaluation

We collect 256 samples of 16-bar music generated by each model from scratch and evaluate the metrics on the samples. We also evaluate the metrics on real data (testing dataset) for reference. The closer the metric is to the real data, the better the model is.

(1) Chord Progression Similarity: it indicates how the distribution of chord progressions generated by the model fits the real data. Higher value (closer to 1) is better. To implement the metric, we perform chord recognition on each bar in the testing dataset and the generated samples then calculate the frequency of each class of chord sequence (1-gram, 2-gram, and 3-gram, such as C-F-G or Am-Em) occurs in them. Afterward, we calculate the cosine

similarity on the distribution of chord classes between the generated samples and the testing dataset to get the value of this metric.

(2) Grooving Similarity (GS): as described in [9]. It indicates the similarity between the groove vectors of each bar. The higher its value, the more consistent the rhythmic pattern is across the song.

(3) Inter-bar Repetition Pattern (RP): by listening to the generated samples, we feel our model can generate more impressive music compared to the baselines. We hypothesize that one of the reasons is richer repetition patterns in our model's music. To quantitatively verify such hypothesis, we designed the metric to measure the frequency of certain repetition patterns in piano rolls. We search for six classes of patterns, AA, AxA, AxxxA, AAA, ABAB, ABxxAB, where for example, ABxxAB means in a consecutive sequence of 6 bars, the 5th bar repeats the 1st bar, and the 6th bar repeats the 2nd bar.

The steps to calculate this metric are:

A. Select a 16-bar music sample.
B. Get its structural distance matrix D where D[i,j] = D[j,i] is the structural distance between bar i and bar j. The structural distance between bar i and bar j is defined as the mean of harmony distances calculated between 32 timesteps between bar i and bar j. See https://github.com/eri24816/improved-diffusion/blob/main/analysis/metrics/structural_dist.py for more details.
C. For an entry D[i,j] < 2 where i<j, we consider bar j is a repetition of bar i. Thus, we binarize the structural distance matrix with a threshold of 2 into the repetition matrix R, which is defined by R[i,j] = 1 if D[i,j] < 2; R[i,j] = 0 if D[i,j] >= 2.
D. On R, we search for certain patterns of active entries (the value equals to 1) and calculate their frequency of existence. For example, an active entry on the superdiagonal indicates an AA repetition pattern. Two consecutive active entries on the superdiagonal indicates an AAA repetition pattern.

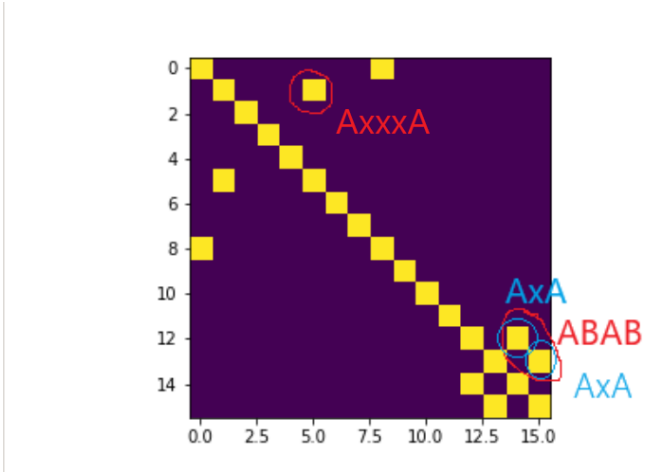

Figure 4. The repetition matrix of a music sample generated by our model. Purple indicates 0, and yellow indicates 1. It contains 1 AxxxA, 2 AxA, and 1 ABAB pattern.

(4) Structureness Indicators (SI): as described in [9]. It measures the richness of repetition patterns in music in granularity of 1-2 bar, 2-4 bar, 4-6 bar, and 6-8 bar. Different from the RP metric, which is measured on piano roll, SI is measured on audio.

4.4.2 Subjective Evaluation

For a listening test, we randomly collect 12 groups of audio samples, where each group is three 16-bar music generated by our model, image diffusion baseline, and CP. 8 participants are asked to rank the 3 audio samples in each group by "how good the audio sample sounds as a piece of pop piano music".

5 Results and Discussion

	Chord Progression Similarity			Grooving Similarity (GS)	Inter-bar Repetition Pattern (RP)						Structureness Indicator (SI)			
	1-gram	2-gram	3-gram		AA	AxA	AxxxA	AAA	ABAB	ABxxAB	1-2 bar	2-4 bar	4-6 bar	6-8 bar
dataset	1.000	1.000	1.000	0.604	0.097	0.128	0.157	0.057	0.069	0.113	0.334	0.261	0.202	0.071
CP	0.958	0.911	0.744	0.426	0.053	0.039	0.029	0.023	0.008	0.007	**0.339**	**0.223**	0.115	0.021
image diffusion	**0.997**	**0.981**	**0.911**	0.556	0.055	0.075	0.111	0.022	0.028	0.056	0.308	0.180	0.105	0.022
video diffusion (ours)	0.996	0.976	0.905	**0.565**	**0.075**	**0.112**	**0.167**	**0.036**	**0.055**	**0.110**	0.323	0.219	**0.141**	**0.039**

Table 1. The result of the objective evaluation.

In summary, our model outperforms both baselines on the metrics of GS, RP, and long-term SI, while it is slightly inferior (< 2% of difference) to image diffusion on chord progression and inferior to CP on short-term SI.

5.1 Strength

The image diffusion baseline, compared to the dataset, suffers lower GS, RP, and SI, which indicates the baseline cannot introduce sufficient patterns across bars. The issue sees significant improvement by switching to video-like representation. This implies that temporal attention does enable the model to generate richer patterns in the music.

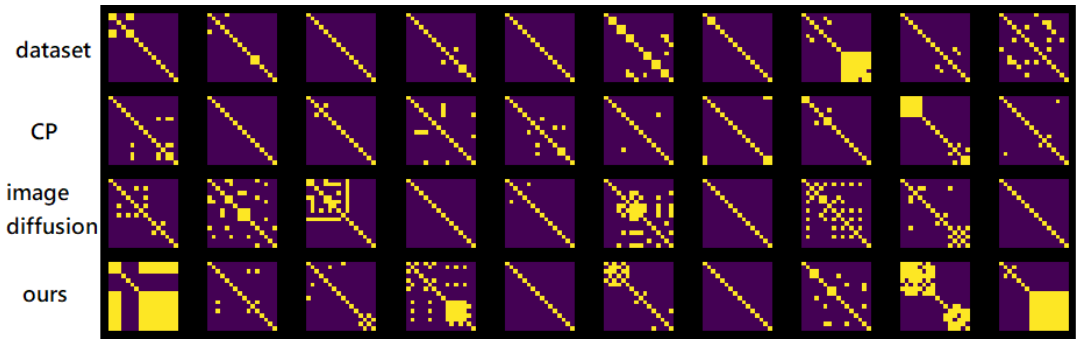

Figure 5. Repetition matrices for samples from the dataset, CP, image diffusion, and our model. Purple indicates 0, and yellow indicates 1.

For metrics of short-term patterns, our model's performance is close to the baselines. However, for metrics of long-term patterns (such as RP-AxxxA, RP-ABAB, RP-ABxxAB, SI 6-8 bars), our model outperforms the baselines significantly. From this, we can tell that temporal attention helps the model learn long-term patterns by reducing the attention distance between distant bars.

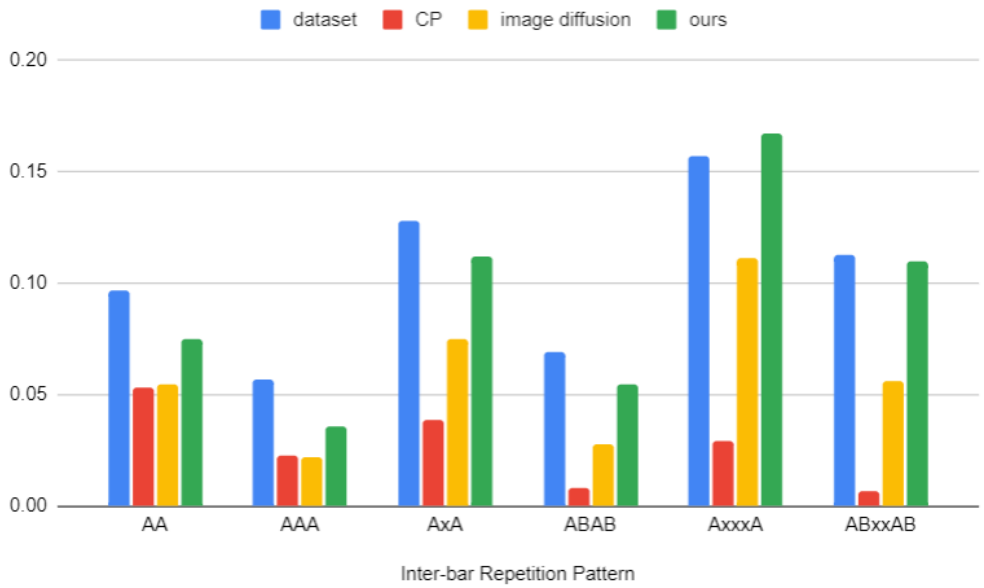

Figure 6. Comparison of all models on the metric of inter-bar repetition pattern.

We believe coherent patterns across bars are a key for music to be impressive to humans, which explains the preference to our models in the user study.

	Wins of our model	Loses of our model	wins/(wins+loses)
Ours vs image diffusion	18	6	0.75
Ours vs CP	20	4	0.83

Table 2. The result of the subjective evaluation. 8 participants are each asked to evaluate 3 groups of samples, resulting in 24 different responses ranking the three models from best to worst.

5.2 Limitations

The fact that CP achieves higher short-term SI scores than both diffusion models implies that Diffusion models have yet to surpass Transformers' inherent strength as an auto-regressive model to construct better structure in a context that is short enough.

Our model sometimes generates music with too many repeats, making the music piece less interesting.

6 Conclusion

In this research, we managed to improve the generation quality of pop piano music by inserting temporal attention blocks into the diffusion model. Compared with the baselines, our model can introduce richer patterns across bars, which is crucial for music to be impressive, interesting, and

close to real human-made music. To further improve the generation quality, future works may try other variations of attention connections or include extra channels for offset, pedal and speed, which have been omitted in this work. Also, with the fast-paced development of diffusion models in recent years, we expect variants of our model with improved diffusion model settings (e.g. the noise schedule) will perform even better on piano music generation.

7 References

[1] Yu-Siang Huang, Yi-Hsuan Yang (2020). Pop Music Transformer: Beat-based Modeling and Generation of Expressive Pop Piano Compositions.

[2] Wen-Yi Hsiao, Jen-Yu Liu, Yin-Cheng Yeh, Yi-Hsuan Yang (2021). Compound Word Transformer: Learning to Compose Full-Song Music over Dynamic Directed Hypergraphs.

[3] Kin Wai Cheuk, Ryosuke Sawata, Toshimitsu Uesaka, Naoki Murata, Naoya Takahashi, Shusuke Takahashi, Dorien Herremans, Yuki Mitsufuji (2022). DiffRoll: Diffusion-based Generative Music Transcription with Unsupervised Pretraining Capability.

[4] Chen Zhang, Yi Ren, Kejun Zhang, Shuicheng Yan (2022). SDMuse: Stochastic Differential Music Editing and Generation via Hybrid Representation.

[5] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, David J. Fleet (2022). Video Diffusion Models.

[6] Gautam Mittal, Jesse Engel, Curtis Hawthorne, Ian Simon (2021). Symbolic Music Generation with Diffusion Models.

[7] Qiuqiang Kong, Bochen Li, Xuchen Song, Yuan Wan, and Yuxuan Wang (2020). High-resolution Piano Transcription with Pedals by Regressing Onsets and Offsets Times.

[8] Alex Nichol, Prafulla Dhariwal (2021). Improved Denoising Diffusion Probabilistic Models.

[9] Shih-Lun Wu, Yi-Hsuan Yang (2022). The Jazz Transformer on the front line: Exploring the shortcomings of AI-composed music through quantitative measures.