The Effect of Explicit Structure Encoding of Deep Neural Networks for Symbolic Music Generation

Ke Chen[1 4], Weilin Zhang[2 4], Shlomo Dubnov[3], Gus Xia[4] and Wei Li[1]
[1]School of Computer Science, Fudan University, China
[2]Department of Computer Science, University of Illinois Urbana-Champaign, USA
[3]Department of Music, University of California San Diego, USA
[4]Department of Computer Science, NYU Shanghai, China
kchen15@fudan.edu.cn, weilinz2@illinois.edu, sdubnov@ucsd.edu, gxia@nyu.edu, weili-fudan@fudan.edu.cn

***Abstract*—With recent breakthroughs in artificial neural networks, *deep generative models* have become one of the leading techniques for computational creativity. Despite very promising progress on image and short sequence generation, symbolic music generation remains a challenging problem since the structure of compositions are usually complicated. In this study, we attempt to solve the melody generation problem constrained by the given chord progression. In particular, we explore the effect of explicit architectural encoding of musical structure via comparing two sequential generative models: LSTM (a type of RNN) and WaveNet (dilated temporal-CNN). As far as we know, this is the first study of applying WaveNet to symbolic music generation, as well as the first systematic comparison between temporal-CNN and RNN for music generation. We conduct a survey for evaluation in our generations and implemented *Variable Markov Oracle* in music pattern discovery. Experimental results show that to encode structure more explicitly using a stack of dilated convolution layers improved the performance significantly, and a global encoding of underlying chord progression into the generation procedure gains even more.**

***Index Terms*—symbolic music generation, artificial intelligence, deep generative model, machine learning and understanding of music, Variable Markov Oracle, analysis of variance, music structure analysis**

I. Introduction

Automated music generation has always been one of the principal targets of applying AI to music. With recent breakthroughs in artificial neural networks, *deep generative models* have become one of the leading techniques for automated music generation [1], and many systems have generated more convincing results than traditional rule-based methods [2]. For the examples of re-generating J.S. Bach's work alone, we have seen [3] - [5].

Despite these promising progress, people still struggle to generate well-structured music. It is worth noting that most successful cases of automatic music compositions were limited to Bach, and at least for non-experts the structure of Bach's compositions is rather local and easy to perceive compared to many other composers. In other words, automatic composition remains a challenging problem since music structures, for most compositions, are complicated and involve long-term dependencies. To solve this problem, some studies imposed structural restrictions [6] - [8] on the final output. However, such post-processing restrictions usually conflict with the generating procedure and require tedious parameter tuning in order to make the algorithm converge. It makes more sense to embed the notion of music structure into the model architecture and generative procedure.

In this study, we chose the task in generating melody constrained by given chord progression. As discussed and practiced by [9], the generation of music by computers is considered as music computational creativity. Solving this problem will show the importance of model choices and data representations in *deep generative model*. We did a systematic comparison between two main-stream approaches of handling music structure representation using two sequential representation generative models: LSTM (a type of RNN) and WaveNet (dilated temporal-CNN). The former encodes structure purely implicitly by the memory of hidden states, while the latter adds more explicit structured dependency via a larger receptive field of dilated convolutions. In terms of the dependencies between hidden variables, the relationship between LSTM and WaveNet is analogous to the one between a first-order autoregressive moving average (ARMA) model and a higher-order moving average (MA) model. From a signal processing perspective, the output signals of LSTM and ARMA models depend on both history input and output signals, while the output signals of WaveNet and MA models solely depend on history inputs. To our knowledge, this is the first systematic comparison between temporal-CNN and RNN for symbolic music application.

We focus on symbolic music generation because music structure information is richer at the composition level than the performance and acoustic level [10]. As far as we know, this is the first attempt in applying WaveNet to symbolic music generation (The name of WaveNet implies its usage on audio applications, but in theory the temporal-CNN architecture can also be used for symbolic generation). Similarly to other studies [4] [11], we use chord progression as the global input for both models and turn the task into modeling the conditional distribution of music composition given chords. We present a novel way of encoding chords and melody in a staggered representation. This effectively combines aspect of different time scales of chords and melody in music, learns simultaneously temporal delayed dependencies between melody over past and next two bars, and also learns harmonic-melodic

978-1-7281-1649-5/19/$31.00 ©2019 IEEE
DOI 10.1109/MMRP.2019.00022

simultaneous relations within every two bars of music. Such manipulation makes sense on a real composition scenario since in this context musicians rarely do purely-free improvisation (unconditioned generation) and almost always rely on a predefined guide (e.g. figured bass, chord progression, lead sheet, etc.) which encodes high-level music structure information.

In order to evaluate the performance of the neural model, we conducted a subjective survey to evaluate the quality of generated music. Human judgment takes into account, unconsciously, not only the local musical statistics, but also builds anticipations that keep track of long music structure, such as recognition of salient motifs and their patterns [12]. To date, most of the evaluation metrics for neural music models were done in terms of immediate prediction error, incapable of capturing longer terms salience structures. In order to be able to see how well the neurally generated music is able to learn such structure, we applied an *Information Dynamics* analysis developed by [13] for music pattern discovery. We applied this analysis to several musical music versus model-generated examples. Experimental results show that in terms of *Information Dynamics* ability for encoding of longer terms music structure, using dilated convolution layers improved the performance significantly. Moreover, we found that the results further improve when we incorporated the complete chord progression into the generation procedure rather than merely considering partial past chords. Our results show that repetition patterns will be found more clearly in the generation if we incorporate the global structure into our inputs.

In the next section, we present related works. We describe the methodology in Section III and show the experimental results in Section IV. We discuss several important discoveries in Section V and finally come to the conclusion in Section VI.

II. Related Work

A. WaveNet for Sound Generation

WaveNet [15] was first introduced by Google Deepmind as a generative model for raw audio. Since then, we have seen many follow-up studies. Most works focus on two aspects: improving the speed of WaveNet, and applying WaveNet to audio-related applications. Parallel WaveNet [17] speeds up the generation process, and Fast WaveNet [18] reduces the time complexity. WaveNet was used in many aspects of raw audio generation as auto-encoder and audio synthesizer. Applications include timbre style generator [19], voice conversion [20], speech synthesis [21] [22], speech enhancement [23], cello performance synthesizer [24], and speech denoising [25]. Most convincing results were achieved via adding conditions as an extra input. For example, the neural audio synthesizer by WaveNet auto-encoders [19] add pitch conditioning during training.

B. LSTM for Music Generation

Many music generation works by deep neural networks start with unconditional (monophonic) symbolic melody generation. The initial work [26] implemented the Back-Propagation Through Time (BPTT) algorithm and used melody and duration representation as the training input for generation. Since generation from single melody can be unstructured, follow up works usually includes conditions on chords or other musical features to guide the generation process.

With Recurrent Neural Network (RNN) [27] and its advanced versions (LSTM and GRU) [28] [29] came out, long-term dependency can be captured for music generation. The work by [30] demonstrated that RNNs is capable of revealing some higher-level information in melody generation. They tested the Blues improvisation performance of LSTM by inputting note slices in real time. The work by [31] defined several measurements (Tone division, Mode, Number of Octaves, etc.) and create melody sequences by RNN by varies inspirations. The unit selection method [32] took a series of measures in music as a unit and used a deep structured semantic model (DSSM) with LSTM to predict future units, instead of directly generate essential elements like notes.

An important variation is the bidirectional architecture. DeepBach [4] introduced an innovated bidirectional RNN for music harmonization. However, the main purpose of DeepBach is harmonization, not to use bidirectional neural networks for music generation. This work inspired us to use Bi-LSTM for conditioned melody generation.

III. Methodology

A. Problem Definition

For music piece of length T, given the melody until time point t ($t<T$) and the chords for the whole piece, we aim to generate the melody from t to T under the chord conditioning. In other words, we have two sequences for the input, the melody sequence and the conditioning sequence. During the generation process, the chord condition is given at each time step for a guiding purpose, and the final output shall both keep the melodic flow and interacts with chords.

Such conditioned generation problem is shown in Fig. 1. The black notes $M_{1:t}$ and $C_{1:T}$ represent all the inputs. The blue notes $M_{t+1:T}$ represent the predicted sequence. The upper track is the melody track, and the lower track is the chord track. The conditional probability distribution for one-way model is defined as:

$$p\left(M_{T-t+1:T} \mid C_{1:T}\right)=\prod_{i=1}^{T-t} p\left(m_i \mid m_1, \ldots, m_{i-1}, c_1, \ldots, c_i\right) \tag{1}$$

where m_i is the generated melody at time step i, c_i is the condition at time step i.

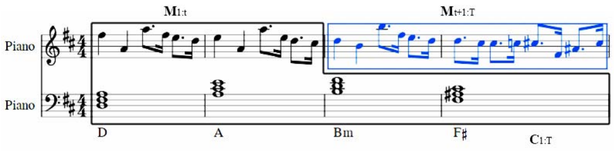

Fig. 1. An illustration of staggered representation in our generation problem

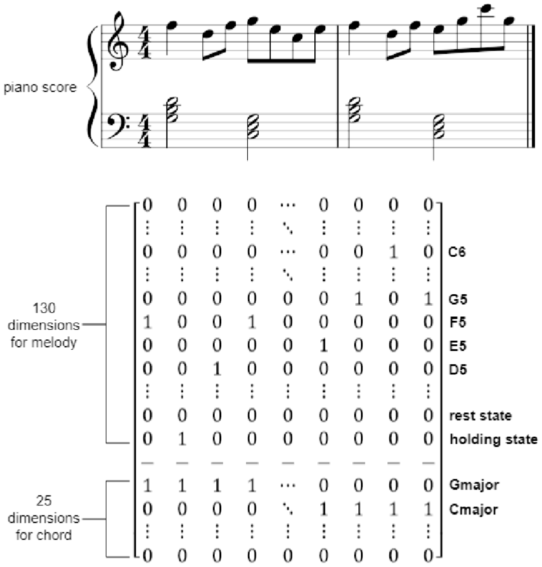

Fig. 2. Data representation of LSTM models.

B. Data Representation

For LSTM, we represent the input as a vector V, which consists of two one-hot vectors M and C, representing melody and chord respectively.

$$V = (M, C) \tag{2}$$

As shown in Fig. 2, the bottom is the original symbolic file in music score. On top of Fig. 2, the two tracks are the representation of vector sequences with melody part and chord part. Note that the signs in the right side do not represent the actual order in the representation. We use a $T \times 155$ dimension vector (130 dimensions for melody and 25 dimensions for chord) to represent a piece of the music with the number of time steps denoted as T. In the melody vector M, the 130 dimensions include pitch, duration, and the rest sign. We use holding state and rest state mentioned in [33]. The first 128 dimensions in M represents pitch value from 0 to 127. Dimension 129 is the rest state, which implies that the note is empty. The last dimension is the holding state, which represents the duration of the previous pitch.

Similar to the melody vector M, the first 24 dimensions in chord vector C represent the most common 12 major chords and 12 minors chords regardless of the inversion (i.e. different root note in one chord). The last dimension is the none chord sign NC. For a chord that is not in the most common 24 chords, we match it to one of the most common chords that share the largest number of same pitches. For example, C-major7 (C7) matches C-major, C-minor7 (Cm7) matches C-minor and C-augment (Caug) matches C-major. The chord vector C does not have the holding state since melody generation is more related to chord value than duration.

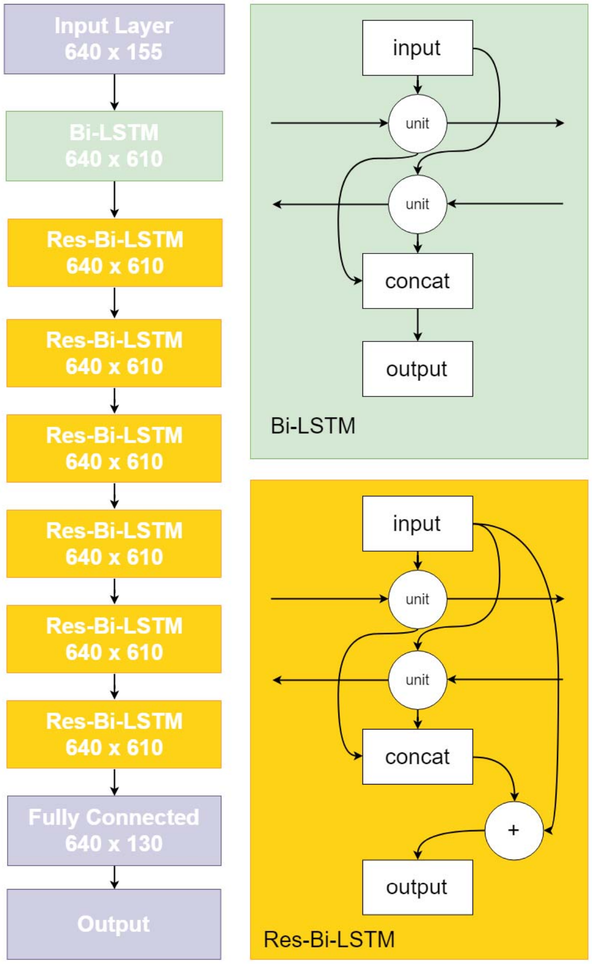

Fig. 3. The bidirectional LSTM model.

For the WaveNet model, the melody vector M has dimension $T \times 128$. Each channel represents a pitch, and the lowest pitch 0 means a rest. If the pitch value is the same in two or more consecutive time steps, we consider them as a sustaining note. For the condition vector, we hash all chords into 24 types, and the chord input is a one-hot vector over the hash table. Each input chord vector C has the shape $T \times 25$ (the additional channels means NC), where the number of time steps T is the same as the corresponding melody vector.

C. LSTM Architecture

In our LSTM model, we start with a one-way model and further develop it into a bidirectional one. As present above, we adapt our data to fit into both the unidirectional model and the bidirectional model.

LSTM is consists of four gates: a cell gate c, an input gate i, an output gate o and a forget gate f:

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \tag{3}$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i), \tag{4}$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o), \tag{5}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \sigma_c(W_c x_t + U_c h_{t-1} + b_c), \tag{6}$$

$$h_t = o_t \odot \sigma_h(c_t) \tag{7}$$

The detailed structure of our bidirectional LSTM model is illustrated in Fig. 3. The left is the input layer followed by 7-layer bidirectional LSTM layers and the fully connected layer. To speed up convergence, we referred to [34] and use the skip-connection in all recurrent layers except the first one, which is shown on the right side.

One thing must be noticed is that the difference between the unidirectional and the bidirectional LSTM model lies in the amount of chord information. In the unidirectional model, we have only previous chord progression when generating the current note. However, the bidirectional model allows us to add the whole chord progression (i.e. the global structure) to the generation procedure. In that, the conditional probability for bidirectional LSTM model should be revised as:

$$p\left(M_{T-t+1:T} \mid C_{1:T}\right) = \prod_{i=1}^{T-t} p\left(m_i \mid m_1, \ldots, m_{i-1}, c_1, \ldots, c_T\right) \tag{8}$$

We should note that the chord condition $c_1, ..., c_T$ is different from the one-way formula $c_1, ..., c_i$. The bidirectional model takes the complete chord progression as the condition.

D. WaveNet Architecture

WaveNet proposed to use a stack of dilated temporal convolution layers [16] for sequential prediction. The original paper also introduced the conditioning feature to guide music generation. For instance, the model can add personal information as a global condition to generate speech from certain people.

We propose to apply WaveNet to symbolic music generation. The unconditioned model with the input melody vector m, and the activation function in dilation layer k is:

$$z = \tanh(W_{f,k} * m) \odot \sigma(W_{g,k} * m) \tag{9}$$

where $*$ represents a dilated convolution operator, $W_{f,k}$ and $W_{g,k}$ are the learnable parameters in the convolution layer, and $\odot$ is a piecewise multiplication operator.

We generate melody with conditioning on chords. We add the embedded chord vector as local conditioning in WaveNet. The activation function at layer k, with the embedded chord condition vector c and the melody vector m:

$$z = \tanh(W_{f,k} * m + V_{f,k} * c) \odot \sigma(W_{g,k} * m + V_{g,k} * c) \tag{10}$$

where the first $*$ in both parentheses represents a dilated convolution operator with $W_{f,k}$ and $W_{g,k}$ as the learnable parameters. The second $*$ in both parentheses represent a 1*1 convolutional layer, with $V_{f,k}$ and $V_{g,k}$ as the learnable parameters. We model the chord conditioning vector as the same length with the melody tensor. In this way, we assign chord to the melody sequence at each time step for monitoring purpose. Conditioning on chords guide the music generation process to include more musical structures, which improves the quality of generated music. The overall architecture and a stack of dilated convolutions is shown in Fig. 4 and Fig. 5.

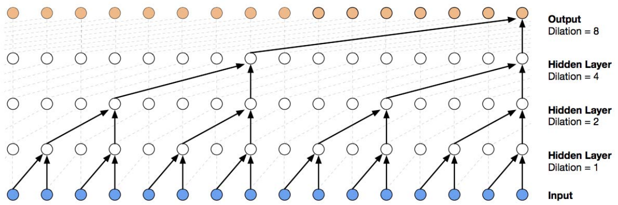

Fig. 4. A stack of dilated temporal convolution layers, reproduced from [15].

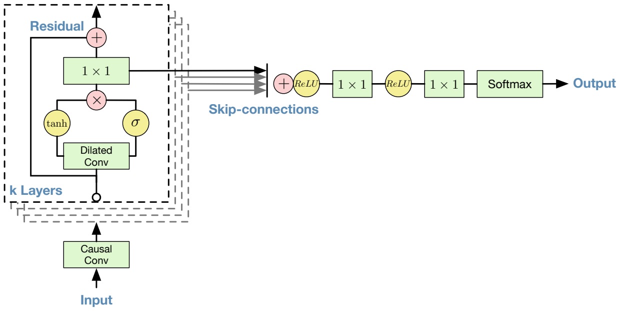

Fig. 5. The WaveNet architecture, reproduced from [15].

For all models, we use the cross-entropy loss between the generation melody and original sample melody as the loss function.

IV. Experiments

A. Dataset

We use the Nottingham Database [35], which consists of 941 folk songs, and each contains both melody and chords. We use 631 songs for training and use data augmentation, switching all the available songs to 12 major(or minor) tonalities, on the training set.

We normalized all the music files to a fixed 120 bpm (beats per minute) to make better alignments in data. We set the frame size to be 1/16 beat(i.e. 0.5/16=0.03125 second) to sample most songs without any quantization error. In the case of triplets, our rule is to set them to be uneven durations. For example, a 5/16 + 6/16 + 5/16 group, we turn such groups back to ternary rhythmic patterns when we convert them to piano score.

B. Survey

We conducted a survey on audiences to compare the performance of the three proposed models (LSTM, Bi-LSTM, and WaveNet). During the survey, each audience listened to 3 groups of music pieces. Each group contains 3 samples generated by 3 models from a same piece of melody (3 x 3 = 9 samples in total). All samples contain 20 beats of original music followed by 20 beats of computer-generated music. After listening to each sample, audiences were asked

to grade it in a continuous scale from 1 (low) to 5 (high). Namely, the final grade considers the following three criteria:
1. Interactivity: Do the chords and melodies interact with each other well?
2. Complexity: Are the notes pattern complex enough to express the theme?
3. Structure: Can you notice some repetitions, forwards, variations in the sample?

C. Hypothesis Test

We performed the ANOVA [36] and two-sample t-test on all pairs of different models.

The null hypothesis of ANOVA is that there is no difference in performance between the three models. Formally:

$$H_0 : \mu_A = \mu_B = \mu_C \tag{11}$$

the alternative hypothesis is that:

$$H_1 : \exists i, j \in \{A, B, C\} : \mu_i \neq \mu_j \tag{12}$$

The null hypothesis of the t-test is that there is no difference in performance between the two test models. Formally:

$$H_0 : \mu_i = \mu_j \tag{13}$$

the alternative hypothesis is that:

$$H_1 : \mu_i \neq \mu_j \tag{14}$$

D. Survey Evaluation

A total of n = 106 people (42 females and 64 males) have completed the survey. 69.81% of them has experiences in music. The aggregated results are shown in Table 1 and Fig. 6. Below is the result table for the within-ANOVA test and the T-tests.

We see that all p-values are smaller than 0.05. Therefore, the performance difference between the three models on conditioned melody generation is statistically significant .

In Fig. 6, the height of the bars represent the means of the ratings, and the error bars represent the mean squared error (MSE).

WaveNet performs better than the unidirectional LSTM model. This result implies that the explicit dependency in dilated convolutions performs better than the implicit dependency in LSTM. The bidirectional LSTM is even better than WaveNet, and in our tests it is the best model. This result shows that embedding future chords in encoder largely improves the model performance. We will discuss the results further in Section V.

E. Pattern Discovery by VMO

We implemented *Variable Markov Oracle* from [13] to illustrate disparities of models in generating music patterns and repetition structures.

TABLE I
RESULT TABLE

ANOVA p-value	3.02e-16
T-test p-value 1, 2	4.10e-16
T-test p-value 1, 3	0.017
T-test p-value 2, 3	5.29e-10

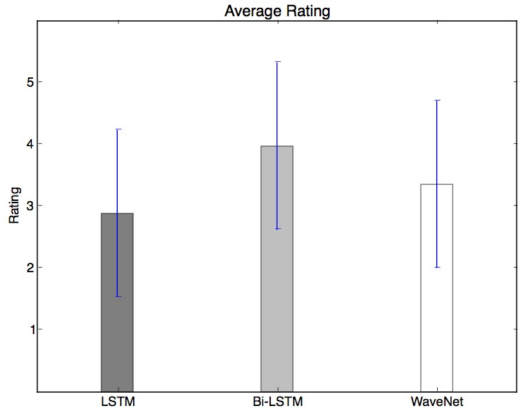

Fig. 6. The subjective evaluation results of the ratings on three models. Different colors represent different models.

The *Variable Markov Oracle* data structure can detect the repeated suffixes, or music patterns within a time series. Given a symbolic sequence $Q = q_1, q_2, ..., q_T$, the VMO carries three kinds of links: forward link, suffix link(*sfx*) and reverse suffix link(*rsfx*). A suffix link of each time state t is the starting point of the longest repeated suffix(*lrs*) of the given Sequence $\{q_1, q_2, ..., q_t\}$. A reverse suffix link is the suffix link in a reverse direction.

As shown in Fig. 7 by [13], an example of the VMO structure in a symbolic signal sequence is provided. Solid arrows represent forward links and dashed arrows are suffix links. The visualization in the bottom part shows how the repetition parts are detected.

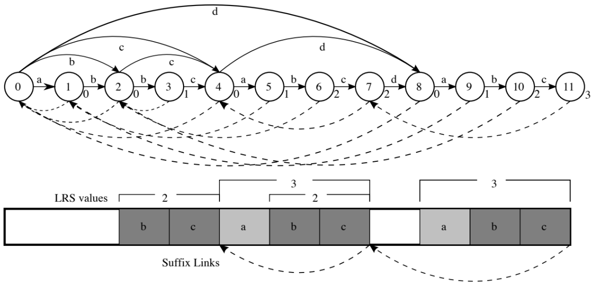

Fig. 7. Reproduced from [13]
(Top) An example of the VMO structure in a symbolic signal sequence {a, b, b, c, a, b, c, d, a, b, c}
(Bottom) A visualization of how patterns {a,b,c} and {b,c} are related to *lrs* and *sfx*

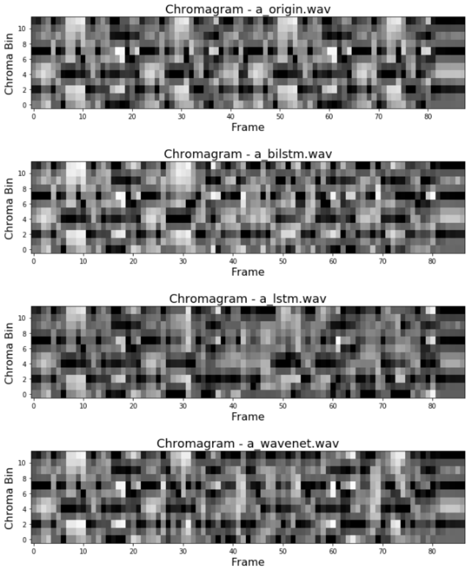

Fig. 8. The chromagrams of samples with the same beginning(10 seconds)

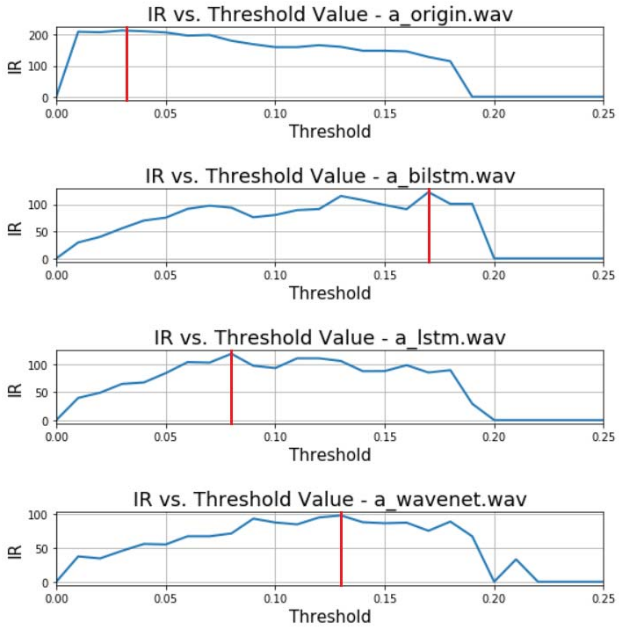

Fig. 9. The *IR*-θ graph in each sample. Red lines denote the threshold we selected in order to maximize the patterns captured in each sample.

We synthesized our generative and original midi files to waves for comparisons. We set the sample rate to 44.1 kHz and implemented *short-time Fourier transform*($stft$) to get the spectrogram. Then the chromagram is obtained by folding the spectrogram into the 12 pitch classes depending on the energy. As shown in Fig. 8, the chromagrams of samples with the same beginning (10 seconds) are provided. From the top graph to the bottom one, each shows the original sample, bidirectional LSTM sample, unidirectional LSTM sample and the WaveNet sample.

We calculated the *Information Rate* (*IR*) [14] to determine the distance threshold θ. Two symbols in a time series O are assigned to be the same if $|O[i] - O[j]| \leq \theta$.

Extremely high or low θ will make VMO incapable of capturing enough patterns. As shown in Fig. 9, the horizontal axis denotes θ and the vertical axis denotes *IR*. A threshold is chosen (red lines) by locating the maximum *IR* value.

Finally, the patterns discovered by VMO in one samples group are shown in Fig. 10. The horizontal axis denotes the time frames and the vertical axis denotes the patterns. Graphs clearly show the disparities in different models. We will discuss these results further in Section V.

V. Discussion

A. VMO Analysis

VMO analysis is able to detect the motifs themselves as repeated patterns of notes, since it finds repetitions using approximately matching suffix search. The plot of these motifs over time (and sub-motifs when the lines overlap vertically) allows visual inspection of such structures. The higher level repetitions also exists in the arrangement of motifs themselves, as shown in Fig. 10 by the boxes. Since each VMO analysis optimizes the threshold of similarity for approximate suffix matching, the motifs shown in the first part of each improvisation appear slightly different. Also, the system takes into consideration also the later motif structure and adjusts its sensitivity so as to produce the most informative representation of the overall information in each piece.

B. LSTM

We found that music generated by LSTM model have great potential in repeating patterns. Fig. 10 shows that the unidirectional LSTM model appears a few repetitions (blue boxes), while the bidirectional LSTM model has more pattern repetitions within the time frames (red boxes and yellow boxes indicate that). Benefit from the short-term memory structure and the explicit input, it is natural for the LSTM model to capture innate structures in the dataset.

Moreover, we noticed that music generated by bidirectional LSTM is more stable and sensitive to chord changes compared to unidirectional LSTM. Fig. 11 shows an example, where the top system represents bidirectional LSTM model and the bottom system represents unidirectional LSTM model. We see that though both models can generate notes segments (of a measure) following the current chord, one-way model fails to take the chord progression as a whole. The generated note sequence tends to be more unstable, smooth, and musical for bidirectional LSTM. This is probably because the generation process includes upcoming chords as a global-structural restriction.

The computational complexity of the LSTM model per layer is $O(nd^2)$, where n is the music sequence length and d is the dimension of the input. Since we implemented the chord progression condition as 25-dimension input, the extra cost of

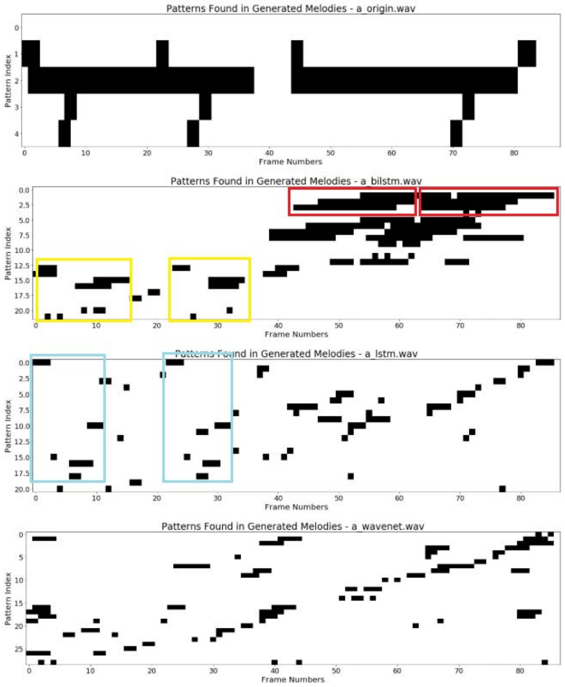

Fig. 10. Patterns discovered in each sample by VMO. The horizontal lines show repetition of individual motifs. Boxes in color show repetitive patterns across motifs detected within a larger time frame.

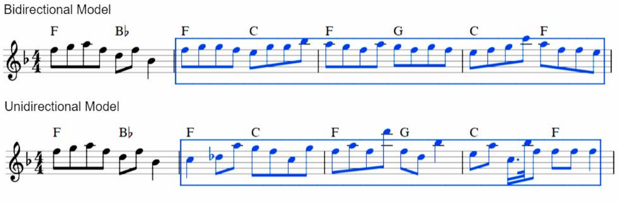

Fig. 11. Comparison of results from one-way and bidirectional LSTM.

our LSTM models is reflected by the increment of dimension d. In practice, we did not see a problem of training efficiency.

C. WaveNet

We found that WaveNet model is able to learn some interesting rhythmic patterns. Fig. 12 shows an example, where the top staff is the (original) input sample, and the bottom staff is the generated notes. We see that both the input and output sequence contain repeated ternary rhythm patterns. Note that the grouping of triplet never breaks up. Since the step size of the generation procedure is very small, the model must have an internal long-term structural representation to capture such a rhythmic pattern.

We argue that such representation comes from the explicit architectural of music structure by the stack of dilated temporal convolution layers. As shown in Fig. 4, the first layer connections can be seen as the rhythmic relationship between 1/16 beat and 1/8 beat, the second layer connections reveal the relationship between 1/8 beat and 1/4 beat, and so on so forth. In other words, the stack of dilated layers happens to agree with the hierarchical rhythmic structure of music composition. At least from the perspective of rhythm generation, WaveNet is more suitable for symbolic music generation than acoustic generation since sounds involve less hierarchical structures.

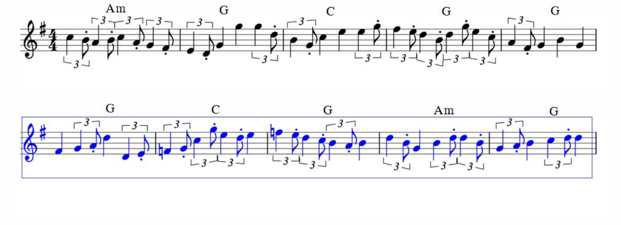

Fig. 12. Generated music learns the rhythm pattern.

Although dilated temporal-CNN can improve the performance significantly rhythmically, it loses some structural features in the real-world music pieces. As shown in Fig. 10, it has neither the obvious repetition throughout the whole music nor the echo among music phrases. Also, unfortunately, the current WaveNet architecture is restricted to one-way music generation. It would be great to develop a bidirectional WaveNet, combining the power of explicit structure modeling and global chord conditions.

The computational complexity of dilated temporal-CNN (WaveNet) per layer is $O(knd^2)$, where k is the kernel size, n is the music sequence length and d is the dimension of the input. Similarly to the analysis of LSTM models, the increment of dimension d by the implementation of chord progression condition will not affect the training efficiency.

VI. Conclusion and Future Work

In this paper, we compared two representative models for conditioned symbolic music generation. In the model design, we first modified WaveNet for symbolic music generation. Then, we proposed bidirectional structures in the LSTM model and further improved the performance. We conducted a subjective evaluation in our experiments, which let subjects to judge the generated samples by interactivity, complexity, and structure. We conducted *Information Dynamics* analysis using *Variable Markov Oracle* in order to capture the effect of different neural models on encoding of longer terms music structure. Such structure is important for human appreciation of music as explained by anticipation theories of music [37]. We were able to analyze the results using motif visualization technique that reveals salient repetition structures in the original versus generated output. The result shows that two critical factors largely improve the model performance: 1) a stack of dilated convolution layers which explicitly encodes the structural dependency of melody sequence, and 2) the incorporation of chord progression as a global structure constraint. In the future, we plan to combine these two factors and develop a bidirectional WaveNet for music generation.

References

[1] J. Briot, G. Hadjeres, and F. Pachet, ”Deep learning techniques for music generation A survey”, CoRR, abs/1709.01620, 2017.

[2] Loy and Gareth, Composing with Computers: A Survey of Some Compositional Formalisms and Music Programming Languages, MIT Press, pp. 291396, 1989.

[3] Liang and Feynman, BachBot: Automatic composition in the style of Bach chorales, University of Cambridge, 2016.

[4] G. Hadjeres and F. Pachet, B DeepBach: a Steerable Model for Bach chorales generation, Proceedings of the 34th International Conference on Machine Learning, PMLR 70:1362-1371, 2017.

[5] C. Z. Huang, T. Cooijmans, A. Roberts, A. Courville and D. Eck, Counterpoint by Convolution, The 18th International Society for Music Information Retrieval Conference, 2017.

[6] S. Lattner, M. Grachten and G. Widmer, Imposing higher-level Structure in Polyphonic Music Generation using Convolutional Restricted Boltzmann Machines and Constraints, Journal of Creative Music Systems, vol. 2, Issue 1, March 2018

[7] P. Verma and J. O. Smith, Neural Style Transfer for Audio Spectograms, 31st Conference on Neural Information Processing Systems, Workshop for Machine Learning for Creativity and Design, 2017.

[8] G. Medeot, S. Cherla, K. Kosta, M. McVicar, S. Abdalla, M. Selvi, E. Rex and K. Webster, StructureNet: INDUCING STRUCTURE IN GENERATED MELODIES, The 19th International Society for Music Information Retrieval Conference, 2018.

[9] D. Cope, Experiments in Music Intelligence (EMI), Proceedings of the International Computer Music Conference, 1987.

[10] S. Dai, Z. Zhang and G. Xia, Music Style Transfer Issues: A Position Paper, Proceeding of International Workshop on Musical Metacreation, 2018.

[11] H. Hermann, F. Johannes and M. Wolfram, HARMONET: A Neural Net for Harmonizing Chorales in the Style of J.S.Bach, Proceedings of the 4th International Conference on Neural Information Processing Systems, pp. 267287, 1991.

[12] A. Cont, S. Dubnov and G. Assayag, Anticipatory Model of Musical Style Imitation using Collaborative and Competitive Reinforcement Learning, Anticipatory Behavior in Adaptive Learning Systems, 2006.

[13] C. Wang, J. Hsu and S. Dubnov, Music Pattern Discovery with Variable Markov Oracle: A Unified Approach to Symbolic and Audio Representations, The 16th International Society for Music Information Retrieval Conference, 2015.

[14] S. Dubnov, G. Assayag and A. Cont, ”Audio Oracle Analysis of Musical Information Rate”, The 5th IEEE International Conference on Semantic Computing (ICSC), pp. 567571, 2011.

[15] A. Oord et al., WaveNet: A Generative Model for Raw Audio, The 9th ISCA Speech Synthesis Workshop, 2016.

[16] F. Yu and V. Koltun, Multi-Scale Context Aggregation by Dilated Convolutions, The 4th International Conference on Learning Representations, 2016.

[17] A. Oord et al., Parallel WaveNet: Fast High-Fidelity Speech Synthesis, Proceedings of the 35th International Conference on Machine Learning, PMLR 80:3918-3926, 2018.

[18] T. L. Paine et al., Fast Wavenet Generation Algorithm, CoRR, abs/1611.09482, 2016.

[19] J. Engel et al., Neural Audio Synthesis of Musical Notes with WaveNet Autoencoders, The 34th International Conference on Machine Learning, 2017.

[20] K. Kobayashi, T. Hayashi, A. Tamamori and T. Toda, Statistical Voice Conversion with WaveNet-Based Waveform Generation, Interspeech, 2017.

[21] Tamamori et al., Speaker-Dependent WaveNet Vocoder, Interspeech, 2017.

[22] J. Shen et al., Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions, IEEE International Conference on Acoustics, Speech and Signal Processing, 2018.

[23] K. Qian et al., Speech Enhancement Using Bayesian Wavenet, Interspeech, 2017.

[24] M. Rachel, T. Vijay, S. Ali and K. Brian, An end to end model for automatic music generation: Combining deep raw and symbolic audio networks, 2018.

[25] F. G. Germain, Q. Chen and V. Koltun, Speech Denoising with Deep Feature Losses, CoRR, abs/1806.10522, 2018.

[26] Todd, A Connectionist Approach to Algorithmic Composition, Computer Music Journal, vol. 13 , 1989.

[27] Z. C. Lipton, A Critical Review of Recurrent Neural Networks for Sequence Learning, CoRR, abs/1506.00019, 2015.

[28] H. Sepp and S. Jurgen, Long short-term memory, Neural computation, MIT Press, vol. 9, pp. 17351780, 1997.

[29] K. Cho et al., Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation, The Conference on Empirical Methods on Natural Language Processing, 2014.

[30] D. Eck and S. Jurgen, Learning the Long-Term Structure of the Blues, Artificial Neural Networks ICANN 2002, Springer Berlin Heidelberg, pp. 284289, 2002.

[31] C. Andres, R. Roseli and Z. Liang, Generation of composed musical structures through recurrent neural networks based on chaotic inspiration, Proceedings of the International Joint Conference on Neural Networks, pp. 32203226, 2011.

[32] B, Mason, W. Gil and H. Larry, A Unit Selection Methodology for Music Generation Using Deep Neural Networks, the 8th International Conference on Computational Creativity, 2017.

[33] A. Roberts, J. Engel and D. Eck, Hierarchical Variational Autoencoders for Music, The 31st Conference on Neural Information Processing Systems, Workshop for Machine Learning for Creativity and Design, 2017.

[34] K. He, X. Zhang, S. Ren and J. Sun, Deep Residual Learning for Image Recognition, IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[35] Nottingham Music Database, http://abc.sourceforge.net/NMD/, 2013.

[36] S. Vineeta, R. R. Kumar and S. Richa, Analysis of repeated measurement data in the clinical trials, Journal of Ayurveda and integrative medicine, Elsevier, vol. 4, pp. 77, 2013.

[37] D. Huron, Sweet Anticipation: Music and the Psychology of Expectation, A Bradford Book, 2008.