Music Composition with Collaboratory AI Composers

Haekwang Kim and Younghwan You
Dept. of computer engineering, Sejong University

Summary

This paper describes an approach of composing music with multiple AI composers. This approach enriches more the creativity space of artificial intelligence music composition than using only one composer. This paper presents a simple example with 2 different deep learning composers working together for composing one music. For the experiment, the two composers adopt the same deep learning architecture of an LSTM model trained with different data. The output of a composer is a sequence of notes. Each composer alternatively appends its output to the resulting music which is input to both the composers. Experiments compare different music generated by the proposed multiple composer approach with the traditional one composer approach.

1. Introduction

Recently, deep learning has been quasi-omnipresent tool for all areas of artificial intelligence problem solution. For creative content generation purposes, there have been researches on deep learning architectures of RBM (Restricted Boltzmann Machine), VAE (Variational Autoencoder), RNN (Recurrent Neural Network), GAN (Generative Adversarial Network) and Transformers, etc.

All these approaches have been used for music generation. A created content, whatever it is, a poem, a music, or an image or a video is something which has regularities or patterns in it. Any content is treated as multi-dimensional numbers (or tensors in deep learning neural networks) in computer and good contents are not any random tensors.

Music theory is a disciplinary that studies regularities for a good music such as chord progression, scales, note intervals and rhythm. A music can be thought as some of tensors in a very high dimensional space that is something to human. Music generation is a process of randomly finding those sample tensors in the sample space. So, artificial music composer can be modeled as a probability distribution of music.

Deep learning architectures of RBM (Restricted Boltzmann Machine), VAE (Variational Autoencoder), RNN (Recurrent Neural network), GAN (Generative Adversarial Networks) and Transformers are all modeling the probability distribution of music contents by training and the deep learning decoder is producing a sample music according to the probability distribution, when some condition is modeled together, then conditional probability distribution is modeled.

Different condition will generate different music. There are music generation deep learning networks such as MuseNet [1] (Transformer architecture) by OpenAI, MuseGAN (GAN architecture) [2] by Academia Sinica and MuseVAE [3] (VAE architecture) by Google.

This paper presents an approach of composing music with multiple AI composers with a simple example of 2 different deep learning composers working together for composing one music.

In section 2, the proposed architecture is presented with example implementation. In section 3, the experiment results are shown with discussion. Section 4 concludes the paper with future extension of research.

2. Music composition with 2 AI composers

A LSTM based RNN in [4] is used for the proposed music generator. The LSTM is trained with note sequences for predicting the next note with the previous notes with window size of N. N is set to 100. The duration of notes is removed for simplicity. A note in the note sequence is either a note or chords. The input of the LSTM is a midi file. Music21 library [5] is used to process the midi file symbols into note sequences. Music21 is a powerful tool for music theory studies in Python language. The LSTM is trained with bunch of midi files and the trained LSTM generates a note sequence with an initial sequence of N notes.

Fig. 1 shows note sequency generation of the LSTM music composer. **pattern**$_t$ is a sequence of N notes at time t. The LSTM generates one note output **note**$_t$ with this input. At time t+1, **pattern**$_{t+1}$ is input to the LSTM and **note**$_{t+1}$ is generated. **pattern**$_{t+1}$ is constructed by appending **note**$_t$ to the **pattern**$_t$ and deleting the first element of the **pattern**$_t$, keeping the size of **pattern**$_{t+1}$, N, the size of input to the LSTM.

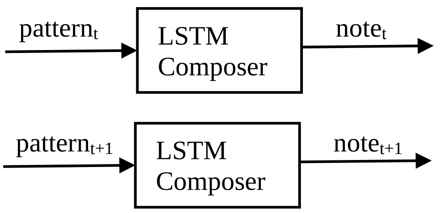

Figure 1 generation of notes from an LSTM music composer

Figure 2 shows the architecture of combining Two LSTM composers for generating one music.

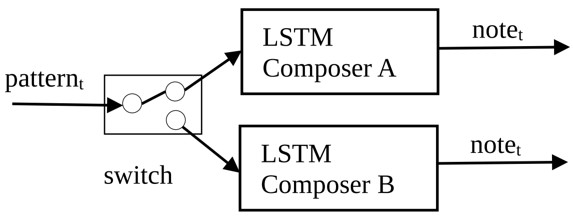

Figure 2 generating one common music from switching two LSTM composers.

pattern$_t$ is a sequence of N notes at time t. one of the LSTM generates one note output **note**$_t$ with this input. At time t+1, **pattern**$_{t+1}$ is constructed in the same manner as single LSTM composer case and one of the LSTM generates one note output **note**$_{t+1}$. One simple method for selecting a composer is by switching with fixed number of output notes.

3. Experiment Results

Total of 92 midi files are used for training the LSTM music generator in [4]. These mid files are grouped into A music and B music groups without any consideration of musical property. Two LSTM composers are obtained by training the LSTM with A music and B music groups. Figure 3 shows the beginning part of a midi file “traitor” for training group A music.

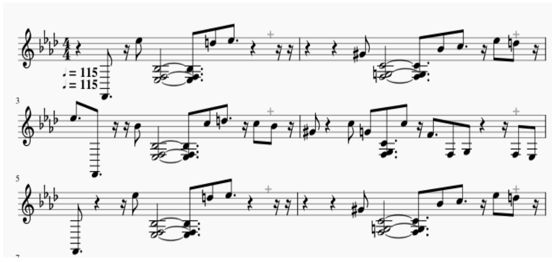

Figure 3 one sample score for training midi file

Figure 4 shows the training accuracy of the LSTM with music group A. The training is done only with training data without validation or test data.

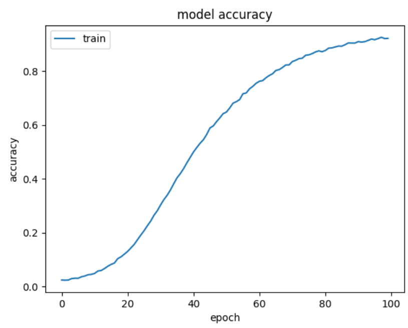

Figure 4 the progress of training with accuracy

Three methods are compared for the experiments with Composer A only, Composer B only and switching Composer A and B. For the switching, the composer is switched at every other 4 note generation. Figure 5, 6 and 7 show the scores generated with the same input pattern by Composer A only, Composer B only and switching LSTM composer A and

LSTM composer B.

Figure 5 music generated by Composer A only

Figure 6 music generated by composer B only

Figure 7 music generated by switching composer A and composer B

The midi files generated from these scores are listened. The midi file from composer A is a bit heavy while the one from the composer B is lighter and the music from switching composer A and composer B is a mixed style of both composers.

4. Conclusion and future work

A simple collaboration of two LSTM music generators is presented for generating one music. Two music generators are obtained by training with different music sources with the same neural network model. There are many areas of research for collaboration of music composer neural networks. First, each music composer needs to be improved for having a specific trait of composing with carefully designed training music dataset. Second, other advanced neural network architectures such as Transformers will be explored to achieve better internal structure of music. Third, the note representation will convey not only the pitch, but be extended to have dynamics and duration. This immense dimension of extended note representation needs to overcome the computing resource problems.

Acknowledgement

This work is supported by “연구역량강화사업” no. 20190505, Sejong University.

References

[1] Alexandru Topirceanu, Gabriel Barina, and Mihai Udresc, “MuSeNet: Collaboration in the Music Artists Industry”, In Proceedings of the 2014 European Network Intelligence Conference (ENIC '14). IEEE Computer Society, USA, 89-94.

[2] Dong, H.-W., Hsiao, W.-Y., Yang, L.-C., and Yang, Y.-H., “MuseGAN: Multi-track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment”, Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI '18), USA, 34-41.

[3] Adam Roberts, Jesse Engel, Colin Raffel, Curtis Hawthorne, Douglas Eck, “A Hierarchical Latent Vector Model for Learning Long-Term Structure in Music”, Proceedings of the 35th International Conference on Machine Learning, 4364-4373, 2018.

[4] Sigurður Skúli, “How to Generate Music using a LSTM Neural Network in Keras”, Towards data science Blog, https://towardsdatascience.com/how-to-generate-music-using-a-lstm-neural-network-in-keras-68786834d4c5, Dec 7, 2017.

[5] Cuthbert Michael Scott, Ariza Christopher, Downie, J. Stephen, Veltkamp, Remco C, “Music21: A Toolkit for Computer-Aided Musicology and Symbolic Music Data”, International Society for Music Information Retrieval, 2010, 637-642