A Novel Method of Music Generation Based on Three Different Recurrent Neural Networks

Jiatong Xie*

The Webb Schools, Claremont, California, USA

*Corresponding author e-mail: hzhe_wjdi@163.com

Abstract. Nowadays, music generation has become an important research field in the study of deep learning. Many different methods such as Generative Adversarial Networks (GAN) and Variational self-encoding (VAE) are already proved to be able to handle such task. In this paper, RNNs, such as LSTM and GRU, will be used to compose the core of the network to predict notes of the melody and generate new music. MIDI will be used as the general music format because of its simple and appropriate data structure that can be coded as a dictionary.

1. Introduction

With the development of various digital techniques, music, as a traditional form of art, has become increasingly significant for people’s lives. In a society with fast pace and fierce competition, many feel stressed and suffer from mental diseases. Research in neuroscience found that some music can change our mood, soothe tired brains and help with the treatment of some severe diseases, such as anxiety, schizophrenia, and even tumors. One of the main purposes of this music generation project is to establish a model that can be useful in music therapy in the future.

This paper evaluates and verifies music generation with different network structures, such as long short-term memory network (LSTM), recurrent neural network (RNN) and Gated Recurrent Units (GRU) [12]. Three different kinds of recurrent neural networks are used to predict every following note in the music through network training combined with dictionary coding of MIDI data set. The model uses MIDI instead of mp3 because the size of the MIDI file is comparatively smaller, which results in a faster training process. We construct the generation model through the corresponding weights, and the model completes the automatic generation of the next note by inputting the first note, achieving the goal of music generation.

2. Methods

In this section, we will discuss word embedding technique, Dropout Layers and three different recurrent networks, RNN, LSTM, GRU. The whole network model for music generation is comprised of all these elements, and thus, it is necessary for us to demonstrate these models and techniques in the first place.

2.1. Word embedding

Word embedding is a technique that gives a name to a word in other forms, as exemplified by the Word2vec approach [13]. It is a general name for a set of language modeling and feature learning techniques in natural language processing (NLP), where words or phrases from the vocabulary are
mapped to the vectors of real numbers. In other words, Word Embedding, or Distributional Vectors, is a technique that converts natural language words into vector or matrix forms which computers can understand.

2.2. Dropout Layers
In the model of machine learning, if there are too many model’s parameters and not enough training samples, the training model can lead to a phenomenon called over-fitting. Over-fitting can be seen in a circumstance where the training data might appear as good, though the actual result might not.

Overfitting is a common issue in machine learning. If one model is over-fitted, the result produced can hardly be used. In order to solve the problem, people usually use the method of model integration, that is, training multiple models to combine together. However, training the combination of models is time-consuming, and testing them can also waste a lot of time.

![Figure 1. A Brief display of Dropout Strategy](image)

2.3. Recurrent Neural Networks
The experiment uses the recurrent neural network (RNN), which is a neural network that can learn a series of items. RNN records time series data and is trained to predict the next element of a sequence. Thus, people prefer to use RNN when they process music and video. RNN predict the following element by reentering the output of the hidden layer (ht) as an additional input, and it can learn based on not only the current information but also its previous state. RNNs are routinely used for natural text processing and for music generation [17].

![Figure 2. RNN neural network structure](image)

2.4. Gated Recurrent Units (GRU)
Gated Recurrent Units (GRU) was first proposed by Cho et al in 2014 [19]. It is similar to LSTM, but simpler to compute and implement because it minimizes the number of gate units. GRU significantly decreases the training time but ensures accuracy at the same time. Its structure of cells and calculation formula can be seen in Fig.4.
In this section, we demonstrate the main components of our model that train and generate music. In the next chapter, we will outline the preparation process of the music data set, as well as the overall structure of the network, and complete the preparation process of the experiment.

3. Experiment preparation

3.1. Networks Structure
In the second chapter, we have introduced the main components and techniques that will be used to compose our networks. The main structure of the network is shown in Fig. 4.

![Figure 3. the structure of GRU NN cell and its calculation formula](http://blog.csdn.net/11010101)

![Figure 4. The Main structure of the network.](http://blog.csdn.net/11010101)
In Fig. 4, we can see that the recurrent neural network we used here is LSTM, which can be substituted by GRU and the most basic RNN. The first layer of the model is an embedding layer. Three LSTM and Dropout layers are constructed after the embedding layer. Then, a dense layer follows, and after the dense layer, we used a SoftMax Layer as the activation layer of the network’s output. The SoftMax Layer generates a pointer, which points to the corresponding char in the notes and chords’ dictionary that will be discussed later. SoftMax layer can be used to predict next notes or chords in a piece of music. By entering an original dataset and generating next notes or chords recurrently, the model is able to generate new music.

3.2. Dictionary code
Dictionary code is important in this paper, which transforms notes and chords, also called char data, into a digital form that can be utilized in the network. Music21, which has been introduced above, is able to read MIDI files and obtain notes and chord’s information from the files. We should first store all of these notes and chord’s information so that they will be composed into a char list. For example, there is a song called Loser by Bigbang in our music dataset. After being processed, it can be presented as ['G5', '10.3', 'B-4', 'G5', 'G5', 'B-4', 'F5', ... , 'E-5', 'F4', 'D4', '5.10']. The information above represents all of its notes and chord information. This can be used to process every song in our datasets and combine them into one list as shown in Fig. 5.

Reading MIDI Files
Processing file : ./data\adele_-_Hello.mid
Processing file : ./data\Ai_ga_Mebraeru_Piano_-_Highschool_DxD_OST.mid
Processing file : ./data\Avril_Lavigne_-_Give_You_What_You_Like.mid
Processing file : ./data\Avril_Lavigne_-_When_Youre_Gone_piano_solo.mid
Processing file : ./data\Big_Bang_-_Last_Dance (_DJS137).mid
Processing file : ./data\Big_Bang_-_Loser (DJS 137).mid
Processing file : ./data\Big_Bang_-_Sober (DJS 137).mid
Processing file : ./data\Colors_-_Halsey.mid
Processing file : ./data\Date_2__Mitsuhhas_Theme_Kimi_no_Na_wa.mscz.mid
Processing file : ./data\DON_T_KNOW_WHAT_TO.DO_-_BLACKPINK_Piano.mid
Processing file : ./data\Ed_Sheeran_Perfect_Piano_Cover.mid
Processing file : ./data\Hellevator_-_JYP_Stary_Kids.mid
Processing file : ./data\Hiyoku_no_Hane_-_Yosuga_no_Sora_Full.mid
Processing file : ./data\If_You_-_Bigbang.mid
Processing file : ./data\IKON_Love_Scenario.mid
Processing file : ./data\Kataware_Doki_-_Kimi_no_Na_wa.mid
Processing file : ./data\Kenshi_Yonezu___Stern.0.mscz.mid
All files read. Daming notes to file
Total Time Taken : 0:00:02.521290

(a)
Figure 5. Figure (a) represents the transforming process, which transforms the MIDI files to char lists, and Figure (b) displays the transforming results.

4. Experimental result and analysis

4.1. Training Result

The graphs of loss and accuracy in Fig. 6 compare three networks, GRU, LSTM and RNN. GRU seems to produce the best result out of three. LSTM has the lowest accuracy and highest loss at the very beginning but eventually becomes the second place. The most basic RNN seems to be unstable, and its final accuracy is the lowest out of all three. However, the overall training process is good and ideal because all three networks reach at least 80% of accuracy in the experiment.
Figure 6. Figure (a) represents prediction accuracy and Figure (b) represents the loss.

4.2. Music Generation Result
After training the model, we save our music products every 50 epoch numbers until 500 epochs. The length of the music is also controllable. The computer is able to generate music in the MIDI form, and by placing the file in a synthesizer can change the elements or form of the music. Even though overfitting does not occur in the process, we cannot ensure that the piece of music with 500 epochs sound absolutely the best because the evaluation of music can be extremely subjective. I eventually choose a piece with 350 epochs based on my own personal preference, and the two figures below show the score and piano roll of the song.

Figure 7. Score of the music
Because music is in MIDI form, it can be easily changed based on people’s needs. For example, if we slow the tempo of the song, it becomes very soothing and relaxing. But if we fasten the tempo, it becomes a cheerful and pleasant song. People can also change the instrument of the song to see different effects.

5. Conclusion
This project mainly evaluates music generation using three distinctive recurrent neural networks: RNN, LSTM, and GRU. The recurrent neural networks are able to record time series data and predict notes based on previous information. We sample every song in MIDI form to minimize the time for the model to practice and to use dictionary coding to represent the notes. With no overfitting and all three networks’ accuracy over 80%, the result of the experiment meets our expectation and is the ideal situation of music generation.

The development in music generation using digital techniques is not only significant for musicology and computer science but can also be applied to many different areas such as psychology and neuroscience by generating music that is soothing for human’s brains. While it is hard for music composers to deliberately write music that is good for improving mental health, computers can automatically generate music for the therapy simply by selecting several samples. The model can serve as a perfect tool for music therapy since the tempos and instruments of the music can be alternated easily by the phycologists.

With the help of music generation, many current issues can be solved, and creative ideas can become reality. The next step is to put more attention on GRU as the main approach of music generation. It is important to accelerate the training process and minimize the size of the dataset for generating music more effectively in the future.

References
[1] Wu Fei, Zhu Wen-wu, Yu Jun-qing. Multimedia technology research:2014—deep learning and media computing[J]. Chinese Journal of Image and Graphics, 2015, 20(11): 1423-1433
[2] Ke Deng-feng, Yu Dong,Jia Jia. Guest editorial for special issue on deep learning for speech, text and image understanding[J]. Acta Automatica Sinica,2016, 42 ( 6) : 805-806
[3] Baum L E , Petrie T , Soules G , et al. A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains[J]. The Annals of Mathematical Statistics, 1970, 41(1):164-171.
[4] Sumi K, Arai M, Fujishima T, et al. A music retrieval system using chroma and pitch features based on conditional random fields[C]// IEEE International Conference on Acoustics. 2012.
[5] Drewes F, Johanna Högberg. An Algebra for Tree-Based Music Generation[J]. 2007.
[6] Andries V D M, Schulze W. Music Generation with Markov Models[J]. IEEE Multimedia, 2011, 18(3):78-85.
[7] Hinton G E, Osindero S, Teh Y W. A Fast Learning Algorithm for Deep Belief Nets[J]. Neural Computation, 2006, 18(7):1527-1554.
[8] Huang Q, Huang Z, Yuan Y, et al. A New Method Based on Deep Belief Networks for Learning Features from Symbolic Music[C]// International Conference on Semantics. IEEE, 2016.
[9] Lyu Q, Wu Z, Zhu J, et al. Modelling High-Dimensional Sequences with LSTM-RTRBM: Application to Polyphonic Music Generation[C]// International Conference on Artificial Intelligence. AAAI Press, 2015.
[10] Choi K, Fazekas G, Sandler M. Text-based LSTM networks for Automatic Music Composition[J]. 2016.
[11] Sturm B L, Santos J F, Ben-Tal O, et al. Music transcription modelling and composition using deep learning [J]. ArXiv preprint ArXiv: 1604.08723, 2016.
[12] Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8):1735-1780.
[13] Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and their Compositional[ity][J]. Advances in Neural Information Processing Systems, 2013, 26:3111-3119.
[14] Y. Bengio, R. Ducharme, P. Vincent. 2003. Neural Probabilitistic Language Model. Journal of Machine Learning Research 3:1137–1155.
[15] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuğlu, P. Kuksa. 2011. Natural Language Processing (Almost) from Scratch. Journal of Machine Learning Research 12:2493–2537.
[16] Hinton G E, Srivastava N, Krizhevsky A, et al. Improving neural networks by preventing co-adaptation of feature detectors[J]. Computer Science, 2012, 3(4):págs. 212-223.
[17] Briot J P, Hadjeres, Gaëtan, Pachet, François-David. Deep Learning Techniques for Music Generation - A Survey[J]. 2017.
[18] Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8):1735-1780.
[19] Cho K, Van Merrienboer B, Gulcehre C, et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation[J]. Computer Science, 2014.
[20] Cuthbert, Michael Scott; Ariza, Christopher T. music21: A Toolkit for Computer-Aided Musicology and Symbolic Music Data, 2010