2021 2nd International Seminar on Artificial Intelligence, Networking and Information Technology (AINIT) | 978-1-6654-1296-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/AINIT54228.2021.00119

Music Generation using Deep Learning with Spectrogram Analysis

Jiangan Chen[1, *, †]
[1]School of Computer Science and Engineering
University of Electronic Science and Technology of China
Chengdu, China
[*] 277953061@qq.com

Rong Du[2, *, †]
[2]School of Physics
Peking University
Beijing, China
[*] durong@stu.pku.edu.cn
[†]These authors contributed equally.

***Abstract*—In the popular music market, the music producer takes a large amount of time and effort to produce one piece of decent music. It would be much more productive and efficient if music generation with the machine was made possible, thus our study. LSTM and GAN are implemented in our study. To analyze the generated music, we used spectrogram analysis. We generated several pieces of music using LSTM and GAN. From the loss curves, we see a promising generation of the two models. Analyzed with a spectrogram, we see the music generated by LSTM is more coherent, while music generated by GAN is more rhythmic. The results we have are relatively good, and the spectrogram analysis part is intelligible and clear. Despite our hardware and computing ability limitations, we still have promising results generated from our models. For future works, we might be able to generate undistinguishable music with little effort.**

Keywords-music generation; LSTM; GAN; spectrogram; deep learning; generative model; MIDI

I. Introduction

For most musical professional, music production is the process of creating, developing, and refining recorded music for presentation. Music production often refers to the whole lifecycle of a piece of music-composition, recording, sound design, mixing, and mastering. Though the definition of music production is broad, every workflow of music production now can be done by digital tools thanks to today's technology. The music producer acts as the director of a movie in a traditional production process by creating a vision for the music material and giving musicians advice on realizing the music. Today more and more artists are choosing to self-produce within lots of genres. A computer and DAW (Digital Audio Workstation) are needed for the music production tools to get started as a producer. Moreover, synths, drum machines, groove boxes, and effects pedals are inspiring equipment to add to producers' setup [1]. For music producers, it could take up to several years to finish a song. Leonard Cohen used three years to write "Hallelujah". For today's professional music producers, the average time taken to produce a song is at least one day, with the producer's mind put to the song [2]. Therefore, it takes the music producer's time and effort to produce one piece of decent music. Plenty of music isn't good enough in the popular music market, and only a small percent of the music could go viral.

As scientists have been elatedly picking up music composing with machine learning, lots of AI music generation applications are developed. For example, Mubert, an AI Music Streaming, is accessible in the Apple APP store or Google store, focusing on listening, generating, and sharing AI-powered music. Upon downloading, open Mubert, and the original music stream will be on. With years spent on research and development of the music generation algorithms, Mubert supports users' training by clicking "likes" and "dislikes" to get the most of personalized music generation. If a short snippet of music is needed to hype content for a vlog, advertisement, or corporate presentation, users can almost instantly generate, save and share a one-minute track [3]. For online resources, led by Payne et al. and released on openAI, MuseNet, a deep neural network, has been developed, generating four-minute musical compositions with 10 different instruments and combining styles from country to Mozart to the Beatles. MuseNet discovers patterns of harmony, rhythm, and style by learning to predict the next token in hundreds of thousands of MIDI files using the same general-purpose unsupervised technology as GPT-2, a large-scale transformer model trained to predict the next token in a sequence [4]. The Google AI team even built and uploaded an online AI platform capable of music generation – Megenta [5]. With the help of a rewritten form of Coconet, a fairly straightforward Convolutional Neural Networks (CNN) with batch normalization and residual connections, they also managed to produce doodles of Bach-styled music clips on March 20th, 2019, to celebrate the birthday of Johann Sebastian Bach [6].

In this study, we used the Nottingham dataset [7, 8] and the ADL piano MIDI dataset [9]. These are datasets of piano pieces from different genres. We trained LSTM and GAN to train and generate several music pieces, and we analyzed the results using spectrogram visualization.

The rest of the paper is organized as follows. Section 2 presents a brief review of related works. Section 3 explains our data processing and models, and Section 4 provides results. Conclusions and future works are discussed in Section 5.

II. Literature Review

Throughout numerous developments, machine learning (ML) or deep learning (DL) approaches have become relatively reliable and feasible in computer music. In music genre classification, Li et al. proposed the Daubechies Wavelet Coefficient Histograms feature extraction method, which captured music signals' local and global information simultaneously by computing histograms on their Daubechies wavelet coefficients achieving significant improvements in accuracy [10]. Li and Ogihara also investigated hierarchical

978-1-6654-1296-4/21/$31.00 ©2021 IEEE
DOI 10.1109/AINIT54228.2021.00119

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on March 11,2023 at 13:53:44 UTC from IEEE Xplore. Restrictions apply.

classification with taxonomies and demonstrated an approach for automatically generating genre taxonomies based on the confusion matrix via linear discriminant projection [11]. In another research, Meng et al. considered temporal feature integration, which combined all the feature vectors in a time window into a single feature vector to capture the relevant temporal information, using a multivariate autoregressive feature model [12]. Moreover, inspired by a model of auditory cortical processing, Panagakis et al. addressed classification tasks with a Support Vector Machine (SVM) in a multilinear perspective when extracting multiscale spectro-temporal modulation features [13]. In 2018, Bahuleyan compared the performance of a deep learning approach, wherein a CNN model was trained end-to-end to predict solely using spectrograms, and four traditional machine learning classifiers that utilized hand-crafted features both from time domain and frequency domain [14]. Besides, it should be noticed that researchers gradually figured out a powerful tool for computer music processing through years of study, namely the spectrogram analysis. For instance, Khunarsal et al. once proposed a singing voice recognition algorithm that automatically recognized the word in a singing signal with background music by using spectrogram pattern matching with an accuracy rate of more than 84% [15]. In the paper of George and Shamir, they discussed a method that worked by converting music samples into spectrograms and extracting content descriptors from them, which could organize the albums by chronological order and musical styles [16]. In addition, Neammalai et al. presented the technique to classify music audio through segmenting and transforming data into spectrograms and then applying image processing methods to find the salient characteristics, which attained a maximum accuracy of 95.77% in the SVM performance [17].

In Hochreiter and Schmidhuber's paper released in 1997, they invented Long Short-Term Memory (LSTM) method for learning long-term dependencies [18], which has been solving abundant previously unsolvable problems, especially about sequences or time series. LSTM-based systems can learn to translate languages, control robots, analyze images, summarize documents, recognize speech and videos and handwriting, run chat bots, predict diseases and click rates, and stock markets, e.g. [19]. At WWDC 2016 Developer Conference, Apple explained how LSTM was improving its iPhone, for example, Siri, in various ways [20]. Sutskever et al. presented a general end-to-end approach to sequence learning that made minimal assumptions on the sequence structure by using LSTMs, whose main result was that on an English to French translation task from the WMT-14 dataset [21], the translations produced by the LSTM achieve a BLEU [22] score of 34.8 on the entire test set, where the LSTM's BLEU score was penalized on out-of-vocabulary words. The LSTM did not have difficulty with long sentences, as well. Through comparison, a phrase-based Statistical Machine Translation (SMT) system achieves a BLEU score of 33.3. Still, when they used the LSTM to rerank the 1000 hypotheses produced by the SMT mentioned above, its BLEU score increased to 36.5 [23]. Additionally, Xu et al. introduced an attention-based model that automatically learns to describe the content of images using standard backpropagation techniques and stochastically by maximizing a variational lower bound with LSTM as decoder [24]. Naturally, as musical features tend to be composed and displayed in time-relevant aspects, it showed great importance and fascinating perspective to combine computer music generation with LSTM. In 2002, Eck and Schmidhuber pointed out that LSTM was a good mechanism for composing music and presented experimental results showing that LSTM successfully learns a form of blues music and can compose novel melodies in that style [25, 26]. Coca et al. tried including an independent melody given by a chaotic composition algorithm to provide an inspiration source to the LSTM model that ran until the degree of melodiousness fell within a predetermined range [27]. While Lyu et al. integrated LSTM for memorizing and retrieving useful history information and Restricted Boltzmann Machine for high dimensional data modelling to compose different musical styles and generate polyphonic music [28]. Moreover, Ycart and Benetos investigated the predictive power of simple LSTM networks for polyphonic MIDI sequences using an empirical approach [29]. Choi et al. studied the results of text-based LSTM for automatic music composition [30]. Besides, Mangal et al. imparted a peek view of distributions of weights and biases in every layer of an LSTM model along with a precise representation of losses and accuracy at each step and batches [31].

Generative Adversarial Networks (GAN) are capable of solving series of similar problems. Combined with reinforcement learning (RL), GAN can be designed for sequence generation models [32]. GAN already excels at existing face synthesis problems with authors attempting to create three players instead of the previous generator-discriminator game [33]. GAN can generate sequential results, according to N. Sadoughi and C. Busso [34]. According to multiple papers, GAN is proved to have remarkable results in the generation model field [35]. Moreover, Dong et al. proposed three models for symbolic multitrack music generation under the framework of GANs, trained the proposed models on over one hundred thousand rock music bars, and applied them to generate piano-rolls of five tracks: bass, drums, guitar, piano, and strings. They showed that the models could generate coherent music of four bars right from scratch [36]. Considering the studies done by others, we believe that GAN is fairly competent when used in music generation.

III. Method

A. Data

1) Data Source

Throughout the research process, we used the Nottingham Music Database (NMD) [7] and the Augmented Design Lab (ADL) Piano MIDI [9]. The NMD is a collection of 1200 British and American folk tunes created by Eric Foxley initially as a personal collection. The database was converted to ABC music notation format and then to MIDI form, which is kindly released on GitHub [8]. As for the ADL Piano MIDI, created by Ferreira et al., it is based on the Lakh MIDI dataset [37] and collected because of the lack of publicly available high-capacity datasets in the symbolic music domain. The dataset includes a total of 11,086 unique piano MIDI files from 18 genres, such as Rock, Classical, Jazz, Blues, and Latin, etc.

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on March 11,2023 at 13:53:44 UTC from IEEE Xplore. Restrictions apply.

2) *Data Processing*

All the music files in our research are in the format of MIDI (musical instrument digital interface), which is the most extensive standard music format in music arrangement. MIDI records music with digital signals of notes, like instructions for pitch and volume. Having uploaded music data to our model, we transform them into a single array of notes encoded as a tuple of pitch and chord.

B. *Models*

1) *LSTM*

LSTM has unique feedback connections different from standard feedforward neural networks. It is capable of processing entire sequences of data like speech or video. A common LSTM unit comprises a cell, an input gate, an output gate, and a forget gate. The cell remembers values over time intervals, and the gates regulate the flow of the information. LSTMs were originally developed to deal with the vanishing gradient problem that could be encountered when using traditional RNNs, so LSTM networks are well-suited to making predictions based on time series data [38].

2) *GAN*

A generative adversarial network (GAN) is a class of machine learning frameworks designed by Ian Goodfellow and his colleagues in 2014. In a generative adversarial network, two neural networks contest each other in a zero-sum game, where one's gain is another's loss. GAN learns to generate data with the same statistics as the training set. Though originally proposed as a form of a generative model of unsupervised learning, GANs have also proven useful in semi-supervised learning and reinforcement learning [39].

C. *Spectrogram Visualization*

As a powerful method in signal processing, spectrogram analysis uses spectrograms, a visual representation of the spectrum of frequencies of a signal as it varies with time, to describe features of the signal pieces and draw conclusions. This study analyzes our generation results through plotting mel-spectrogram, zero-crossing rate (ZCR), and chroma frequency diagrams. Among them, the mel-spectrogram, namely the mel-frequency cepstrum (MFC), is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel-scale of frequency which reflects the perceived feelings of loudness (decibel) of human listeners. The ZCR is the rate at which a signal passes zero points in one frame. The chroma frequency feature closely relates to the twelve different pitch classes. To visualize the spectrogram generation, we turn to librosa [40], a python package for music and audio analysis. Our study provides the building blocks necessary to create music information retrieval systems, specifically MFC, ZCR, and chroma extraction and analysis tools.

IV. Results and Discussions

A. *Lost Curve Analysis*

From Fig.1, we could see the losses of discriminator and generator are diminishing. For the first 2 epochs, the loss of the generator is relatively high, which is typical for the generators in generative adversarial networks. After 2 epochs, both games are decreasing in losses. The losses are 0.6698 and 0.6435, respectively. After all, 4 epochs are trained.

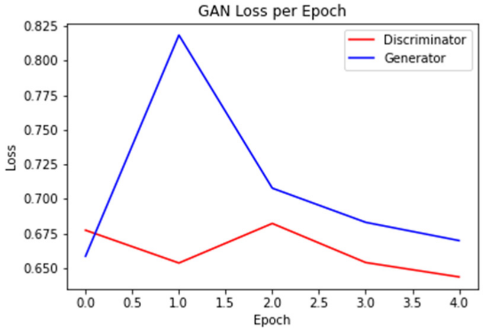

Figure 1. Loss curve of GAN model

Fig.2. shows the loss curve of the training process of the LSTM model. The loss keeps decreasing through the whole training process. The decreasing derivative is relatively higher in the first epoch, and in the epochs after, the slope is relatively lower. At the end of the training process, the loss is 0.6734.

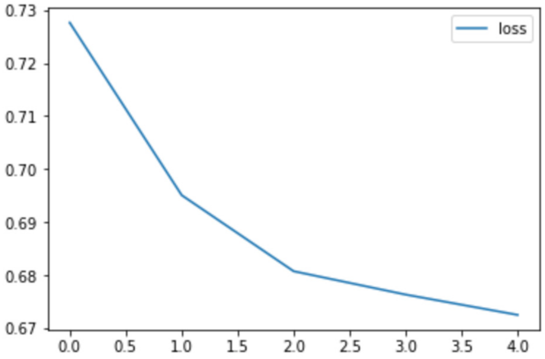

Figure 2. loss curve of LSTM model

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on March 11,2023 at 13:53:44 UTC from IEEE Xplore. Restrictions apply.

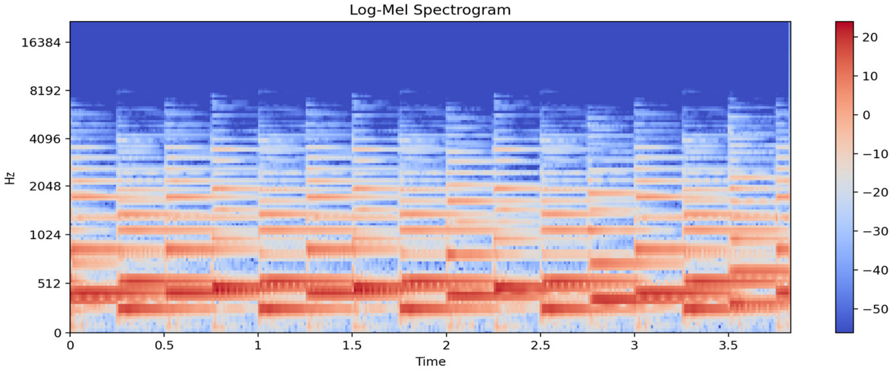

Figure 3. Log-Mel Spectrogram of a result of LSTM model

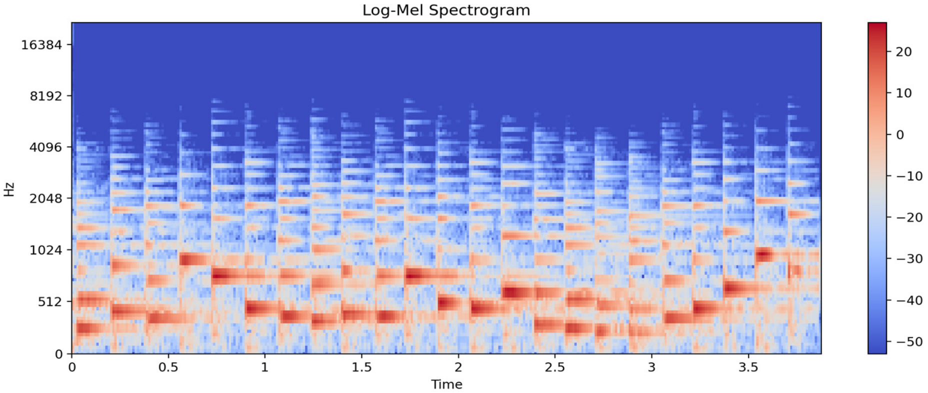

Figure 4. Log-Mel Spectrogram of a result of GAN model

B. *Spectrogram Analysis*

Since we are focusing on music generation, and as even the most developed technical method cannot distinguish deliberate music, the spectrogram analyses we displayed here are mostly straightforward. To present a clearer exhibition, we arbitrarily extract 4 seconds from a randomly chosen music piece by our LSTM and GAN model, respectively, abstracting MFC, ZCR, and chroma features for a diverse perspective.

For MFC, we use a window, which can be interpreted as Fast Fourier Transform windows, with the length of 1024 samples and hop length (the number of samples between successive windows) of 512 samples, indicating a 50% overlap of contiguous windows. We also choose 128 Mel filter banks for the generation of Mel spectrograms, and we print the diagram with a log-scaled (Mel-scaled frequency) vertical axis, which is a fair counterpart with human senses as discussed previously. Therefore, we name the figures as "Log-Mel Spectrogram", which is an appellation common seen in other materials as well. In Fig.3. and Fig. 4., formants are colored in red, which are actually the main constituent of the sound we hear. Even without the assistance of advanced analytic tools, we can tell that the formants of the LSTM production are dense and wide, while formants of the GAN production are sparser and narrower. Thus, the phones, which convergent formants created by the LSTM actually represents, are much more luxuriant and various. Although we cannot comment on the rhythm or melody, the chord by LSTM seems to linger, whereas the chord of GAN appears to be nimble and discrete.

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on March 11,2023 at 13:53:44 UTC from IEEE Xplore. Restrictions apply.

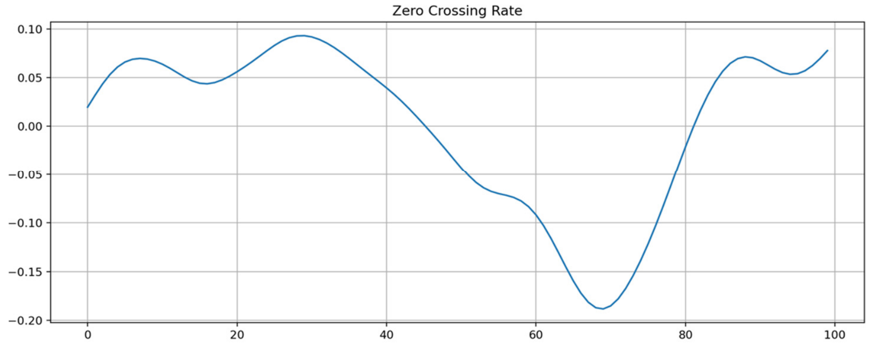

Figure 5. Zero-crossing Rate graph of the result of LSTM model

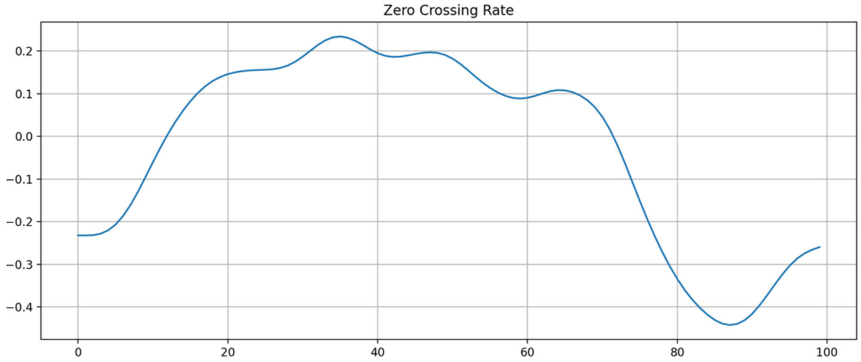

Figure 6. Zero-crossing Rate graph of the result of GAN model

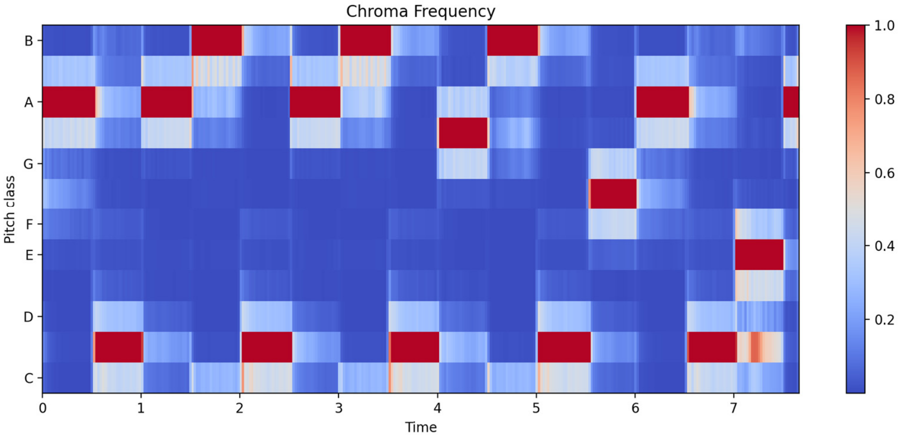

Figure 7. Chroma Frequency graph of the result of LSTM model

Fig. 5 and Fig. 6 present the plots of ZCR. In mechanics, acoustics, to be specific, the amplitude of a sound signal represents its energy proportional to amplitude squared. That is to say, the energy fluctuation, namely the percussive phones or tempo, is more distinguishable with an increasing ZCR. Comparing both graphs with 2 zero-crossing points, we can relate to the Log-Mel that the music by our models have similar rhythmic feature, but the LSTM production is less curvy, which has more duration of notes and hears more soothingly.

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on March 11,2023 at 13:53:44 UTC from IEEE Xplore. Restrictions apply.

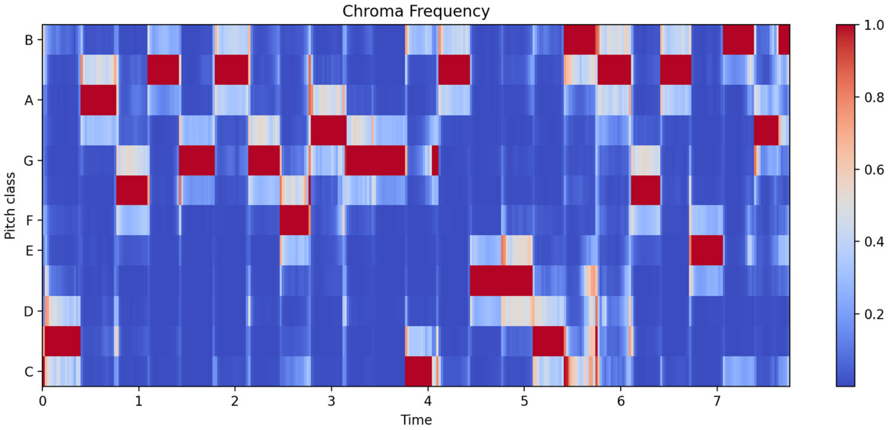

Figure 8. Chroma Frequency graph of the result of GAN model

Unlike ZCR, which describes the frequency shift of the signal, chroma frequency, or sometimes chroma feature, is an indicator of frequency distribution, as shown in Fig. 7 and Fig. 8. It represents the "color" class of a pitch (a property determines on the frequency of a stable signal) with respect to the octave by mapping the frequency-related quality of our spectrogram to a typically 12-dimensional vector ({C, C#, D, D#, …, B}). With the help of librosa's chroma feature tools, we generate chroma frequency graphs using the same pieces of results and the same parameters of MFC. As the red areas are what we actually interpret as pitches, we can clearly distill the chord's harmony and energy distribution of each chord. Three graphs of the two pieces of respective results by LSTM and GAN fairly corroborate each other in stating the relative relieved and ample chord of the LSTM music and the lucid and lively chord of the GAN music.

V. Conclusion

In our research, we have presented two approaches for music generation, which are based on machine learning methods, after witnessing the difficulties in music composing, and the possibilities and potential of music produced artificial-intelligently. We implement an LSTM model and a GAN model, respectively, to enable them to generate classical piano music by feeding the Nottingham dataset and ADL Piano MIDI dataset to our models. Splitting the input data into lists of notes, we make the models learning through predicting the next notes of a given music piece. After several epochs, we apply loss curve analysis on the model and spectrogram analysis, including Mel-frequency cepstrum, zero-crossing rate, and chroma frequency on the pieces of the results. Despite the simplicity of our models and given the limited computing resources we have obtained for the project, we literally complete the whole process of AI music generation and come out with fairly audible music pieces and practical analysis through spectrogram visualizations.

However, our computing ability for model training is relatively low in this study, yet the original database is huge. To solve this problem, we selected a proportion of the original database. The original database contains about ten thousand songs, and training this amount of data on a CPU is quite challenging. So, we cut 90% of the data for the training process. For future work, we could apply GPU or AWS cluster for computing to improve our results. Moreover, the music type we use is limited. In this study, only piano music files are used, which is relatively simple on the genre. We could use more symphony or violin pieces for further study in order for the model to learn more styles of songs. Last but not least, we only used LSTM and GAN models in the current stage of the study. We implement models from keras to complete this task, and only a few default layers are added to our networks. In future work, the models can be modified, and the parameters can be adjusted. Furthermore, reinforcement learning and CNN could be combined with the present models in the next stage.

References

[1] Hahn, M. (2021). Music production: Everything you need to get started. LANDR Blog. Retrieved August 13, 2021, from https://blog.landr.com/music-production/

[2] Duncan, L., Wiebe, D. A., &amp; Letang, S. (2020, December 29). How long does it take to write a song? Retrieved August 13, 2021, from https://www.musicindustryhowto.com/how-long-does-it-take-to-write-a-song/

[3] MUBERT Inc, Google Play. (2021, June 11). Mubert: AI Music Streaming. Retrieved August 24, 2021, from https://play.google.com/store/apps/details?id=com.jellyworkz.mubert&hl=en_US&gl=US

[4] MuseNet. (2019, April 25). Retrieved August 24, 2021 from https://openai.com/blog/musenet/#fn

[5] Magenta. Retrieved August 24, 2021, from https://magenta.tensorflow.org

[6] Huang, C. Z. A., Cooijmans, T., Dinculescu, M., & Hawthorne, A. R. C. (2019). Coconet: the ml model behind today's bach doodle. Accessed July, 9.

[7] Nottingham Database. (2011, October 2). Retrieved June, 2021, from https://ifdo.ca/~seymour/nottingham/nottingham.html

[8] Nottingham Dataset (with MIDI). (2017, March 3). Retrieved June, 2021, from https://github.com/jukedeck/nottingham-dataset

[9] Ferreira, L., Lelis, L., & Whitehead, J. (2020, October). Computer-generated music for tabletop role-playing games. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (Vol. 16, No. 1, pp. 59-65).

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on March 11,2023 at 13:53:44 UTC from IEEE Xplore. Restrictions apply.

[10] Li, T., Ogihara, M., & Li, Q. (2003, July). A comparative study on content-based music genre classification. In Proceedings of the 26th annual international ACM SIGIR conference on *Research and development in informaion retrieval* (pp. 282-289).

[11] Li, T., & Ogihara, M. (2005, March). Music genre classification with taxonomy. In *Proceedings. (ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.* (Vol. 5, pp. v-197). IEEE.

[12] Meng, A., Ahrendt, P., Larsen, J., & Hansen, L. K. (2007). Temporal feature integration for music genre classification. *IEEE Transactions on Audio, Speech, and Language Processing*, *15*(5), 1654-1664.

[13] Panagakis, I., Benetos, E., & Kotropoulos, C. (2008). Music genre classification: A multilinear approach. In *ISMIR* (pp. 583-588).

[14] Bahuleyan, H. (2018). Music genre classification using machine learning techniques. *arXiv preprint arXiv:1804.01149*.

[15] Khunarsal, P., Lursinsap, C., & Raicharoen, T. (2009, June). Singing voice recognition based on matching of spectrogram pattern. In *2009 International Joint Conference on Neural Networks* (pp. 1595-1599). IEEE.

[16] George, J., & Shamir, L. (2014). Computer analysis of similarities between albums in popular music. *Pattern Recognition Letters*, *45*, 78-84.

[17] Neammalai, P., Phimoltares, S., & Lursinsap, C. (2014, December). Speech and music classification using hybrid form of spectrogram and fourier transformation. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific* (pp. 1-6). IEEE.

[18] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

[19] Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, *61*, 85-117.

[20] IOS 10: Siri now works in third-party apps, comes with extra AI features. (2016). Retrieved August 13, 2021, from https://bgr.com/tech/ios-10-siri-third-party-apps-4914313

[21] Bojar, O., Buck, C., Federmann, C., Haddow, B., Koehn, P., Leveling, J., ... & Tamchyna, A. (2014, June). Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the ninth workshop on statistical machine translation* (pp. 12-58).

[22] Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002, July). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics* (pp. 311-318).

[23] Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems* (pp. 3104-3112).

[24] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., ... & Bengio, Y. (2015, June). Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning* (pp. 2048-2057). PMLR.

[25] Eck, D., & Schmidhuber, J. (2002). A first look at music composition using lstm recurrent neural networks. *Istituto Dalle Molle Di Studi Sull Intelligenza Artificiale*, *103*, 48.

[26] Eck, D., & Schmidhuber, J. (2002, September). Finding temporal structure in music: Blues improvisation with LSTM recurrent networks. In *Proceedings of the 12th IEEE workshop on neural networks for signal processing* (pp. 747-756). IEEE.

[27] Coca, A. E., Corrêa, D. C., & Zhao, L. (2013, August). Computer-aided music composition with LSTM neural network and chaotic inspiration. In *The 2013 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-7). IEEE.

[28] Lyu, Q., Wu, Z., Zhu, J., & Meng, H. (2015, June). Modelling high-dimensional sequences with lstm-rtrbm: Application to polyphonic music generation. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

[29] Ycart, A., & Benetos, E. (2017, October). A study on LSTM networks for polyphonic music sequence modelling. ISMIR.

[30] Choi, K., Fazekas, G., & Sandler, M. (2016). Text-based LSTM networks for automatic music composition. *arXiv preprint arXiv:1604.05358*.

[31] Mangal, S., Modak, R., & Joshi, P. (2019). Lstm based music generation system. *arXiv preprint arXiv:1908.01080*.

[32] Guimaraes, G. L., Sanchez-Lengeling, B., Outeiral, C., Farias, P. L. C., & Aspuru-Guzik, A. (2017). Objective-reinforced generative adversarial networks (ORGAN) for sequence generation models. *arXiv preprint arXiv:1705.10843*.

[33] Shen, Y., Luo, P., Yan, J., Wang, X., & Tang, X. (2018). Faceid-gan: Learning a symmetry three-player gan for identity-preserving face synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 821-830).

[34] Sadoughi, N., & Busso, C. (2019). Speech-driven expressive talking lips with conditional sequential generative adversarial networks. *IEEE Transactions on Affective Computing*.

[35] Din, N. U., Javed, K., Bae, S., & Yi, J. (2020). A novel GAN-based network for unmasking of masked face. *IEEE Access*, *8*, 44276-44287.

[36] Dong, H. W., Hsiao, W. Y., Yang, L. C., & Yang, Y. H. (2018, April). Musegan: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[37] Raffel, C. (2016). Learning-based methods for comparing sequences, with applications to audio-to-midi alignment and matching. Columbia University.

[38] Wikipedia contributors. (2021, August 10). Long short-term memory. In *Wikipedia, The Free Encyclopedia*. Retrieved August 24, 2021, from https://en.wikipedia.org/w/index.php?title=Long_short-term_memory&oldid=1046770680

[39] Wikipedia contributors. (2021, August 8). Generative adversarial network. In *Wikipedia, The Free Encyclopedia*. Retrieved August 24, 2021, from https://en.wikipedia.org/w/index.php?title=Generative_adversarial_network&oldid=1037818223

[40] Brian McFee, Alexandros Metsai, Matt McVicar, Stefan Balke, Carl Thomé, Colin Raffel, Frank Zalkow, Ayoub Malek, Dana, Kyungyun Lee, Oriol Nieto, Dan Ellis, Jack Mason, Eric Battenberg, Scott Seyfarth, Ryuichi Yamamoto, viktorandreevichmorozov, Keunwoo Choi, Josh Moore, … Thassilo. (2021). librosa/librosa: 0.8.1rc2 (0.8.1rc2). Zenodo. https://doi.org/10.5281/zenodo.4792298

Authorized licensed use limited to: b-on: UNIVERSIDADE DO MINHO. Downloaded on March 11,2023 at 13:53:44 UTC from IEEE Xplore. Restrictions apply.