International Journal of Innovative Research in Engineering
Volume 4, Issue 3(May-June 2023), PP: 552-554.
www.theijire.com **ISSN No: 2582-8746**

AI Based Music Tunesmith

Amritenra Pratap Singh[1], Shobhit Sharma[2], Amn Singh[3], Ramesh Vaish[4], Anamika Yadav[5]
[1,2,3,5]students, Babubanarsi Das institute of technology and management, LuckNow, Uttar Pradesh, India.
[4]Assistant professor, Babubanarsi Das institute of technology and management, LuckNow, Uttar Pradesh, India.

How to cite this paper:
Amritenra Pratap Singh[1], Shobhit Sharma[2], Amn Singh[3], Ramesh Vaish[4], Anamika Yadav[5], "AI Based Music Tunesmith", IJIRE-V4I03-552-554.

Abstract: *Generating music has some distinct differences from generating images and videos. To begin with, it is an art of time, so a temporal model is essential. Additionally, music is often composed of multiple instruments/tracks with their own tempo, which all unfold over time in a mutually dependent manner. Musical notes are also often grouped into chords, arpeggios, or melodies in polyphonic music, thus introducing a hierarchical ordering of notes. In this paper, we propose three models for symbolic multi-track music generation based on a framework of generative adversarial networks (GANS). The three models differ in their underlying assumptions and corresponding network architectures, and are referred to as the jamming model, the composer model, and the hybrid model. We trained the proposed models on a dataset of more than one hundred thousand bars of rock music, and applied them to generate piano rolls.*

I.INTRODUCTION

Generating realistic and aesthetic pieces is a highly sought-after challenge in the realm of Artificial Intelligence. In recent years, breakthroughs have been made in creating images, videos, and text using Generative Adversarial Networks (GANs) (Goodfellow et al. 2014; Radford et al. 2016; Vendrick et al. 2016; Saito et al. 2017; Yu et al. 2017). There have also been attempts to generate symbolic music, but the task remains difficult due to music's temporal nature. As seen in Figure 1, music has a hierarchical structure composed of higher-level components (e.g., a phrase) made up of repetitious segments (e.g., a bar). People pay attention to structural patterns relating to cohesion, rhythm, tension, and emotion progression.

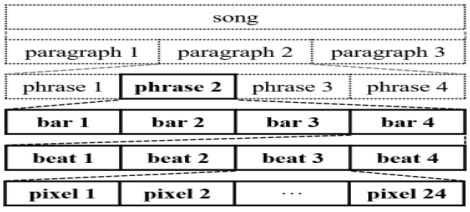


Figure 1: Hierarchical structure of a music piece.

Music is usually composed of multiple instruments and tracks. An orchestra typically consists of brass, strings, wood winds, and percussion instruments, while a rock band often comprises bass, drum set, guitars, and sometimes vocals. These components interact with one another and evolve together. In music theory, there are considerable discussions around composition techniques, such as harmony and counterpoint. Moreover, musical notes are often organized into chords, arpeggios, or melodies. It is not feasible to impose a chronological ordering of notes in polyphonic music; therefore, success in natural language generation and monophonic music generation cannot be replicated in polyphonic music generation. Accordingly, most prior art (as discussed in the Related Work section) has chosen to simplify symbolic music generation in certain ways.

We propose two approaches to incorporate a temporal model: one that creates music from scratch without any human input, and another that learns to follow the given a priori temporal structure of a track. To address the interactions among tracks, we suggest three methods based on our understanding of pop music. The first generates tracks independently using separate generators for each. The second creates them collectively with one generator. The third uses separate generators for each, plus additional shared inputs to guide the tracks to sound harmonious and coordinated. To tackle the grouping of notes, we view bars rather than notes as the basic compositional unit and use transposed convolutional neural networks (CNNs) to generate music one bar at a time, which are known to be effective at locating local patterns.

The core concept of Generative Adversarial Networks (GANS) isto generate adversarial learning by constructing two networks: the generator and the discriminator (Goodfellow et al. 2014). The generator map a random noise, sampled from a prior distribution, to the data space. On the other hand, the discriminator is trained to distinguish real data from data generated by the generator. The generator, on the other hand, is trained to fool the discriminator. The training process can be modeled as a two-player minimax game between the generator G and the discriminator D: min G max D E x sim p d

[log(D(x))]+E x sim p 2 [1 - log(D(G(z)))]. Proposed Model: According to Yang, Chou, and Yang (2017), bar boundaries are often seen as the basic compositional unit for making harmonic changes(eg. chord changes). Furthermore, humansfrequently employ bars asthe fundamental components when writing songs Implementation of Dataset: The piano-roll dataset utilized in this work is derived from the Lakh MIDI dataset (LMD) (Raffel, 2016). We converted the MIDI files to piano-rolls with a height of 128 and width of 96 to capture common temporal patterns. The python library named pretty midi (Raffel and Ellis, 2014) was used to parse and process the MIDI files. Thus, the compiled dataset has been termed as the Lakh Pianoroll Dataset (LPD). Additionally, we present the subset LPD-matched, which is derived from the LMD-matched, an amalgamation of 45,129 MIDIS and entries from the Million Song Dataset (MSD) (BertinMahieux et al., 2011). The conversion utilities, along with the datasets and themetadata, are accessible on theproject website. Analysis of Training Data: The piano-roll dataset utilized in this work is derived from the Lakh MIDI dataset (LMD) (Raffel, 2016). We converted the MIDI files to piano-rolls with a height of 128 and width of 96 to capture common temporal patterns. The python library named pretty midi (Raffel and Ellis, 2014) was used to parse and process the MIDI files. Thus, the compiled dataset has been termed as the Lakh Pianoroll Dataset (LPD). Additionally, we present the subset LPD-matched, which is derived from the LMD-matched, an amalgamation of 45,129 MIDIS and entries from the Million Song Dataset (MSD) (BertinMahieux et al., 2011). The conversion utilities, alongwith the datasets and themetadata, are accessible on theproject website. Training Process: To gain insights into the training process, we examine the generator model used for creating music from scratch (other models have similar behaviors).

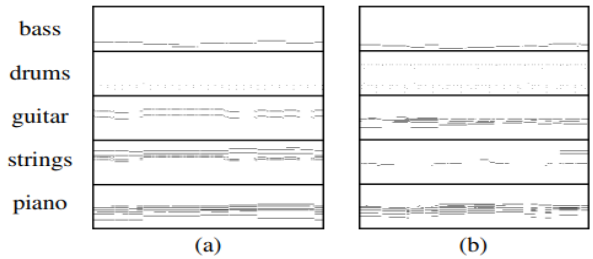
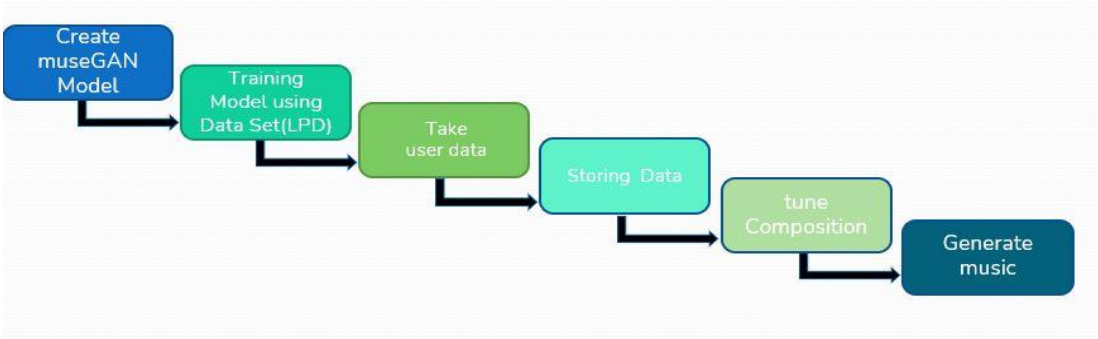


Figure displays the training loss of D as a function of training steps, which is seen to decline rapidly in the beginning and reach a stable level afterward. However, there is also a slight increase in the trend after point B that appears on the graph, suggesting that G commences to learn something from that point onwards. The generated piano-rolls at the five points marked in Figure are shown in Figure .



We can observe how the generated pianorolls develop over the training process. For instance, G grasps the pitch range of each track quite early and starts to produce some notes, though broken up but still within the right pitch range, at point B rather than the noise produced at point A. At point B, we can see clusters of points assembling at the lower part (all with lower pitches) of the bass. After point C, G begins to produce longer segments with a few tones. Related Work: Our model design is based on prior work in video generation that utilize Generative Adversarial Networks (GANs). For example, VGAN (Vondrick, Pirsiavash, &Torralba, 2016) assumed that a video can be broken down into a dynamic © 2023 IJNRD | Volume 8, Issue 2 February 2023 | ISSN: 2456-4184 | IJNRD.ORG IJNRD2302209 International Journal of Novel Research and Development (www.ijnrd.org) c56 foreground and static background, and used 3D and 2D CNNs to generate each component in a two-stream architecture and then combine them with a mask generated by the foreground stream. TGAN (Saito, Matsumoto, & Saito, 2017) used a temporal generator (made of convolutions) to create a fixed-length series of latent variables, which were then fed one by one to an image generator to construct the video frame-by-frame. MOCOGAN (Tulyakov, et al., 2017) assumed that a video can be separated into content (objects) and motion (of objects) and used RNNs to capture the motion of objects. Conclusion: In this paper, we present a new generative model using GANS for multi-track sequence generation. To demonstrate the effectiveness of the model, we have employed deep CNNs to generate multi-track piano-rolls. Additionally, we have designed several objective metrics to gain insights into the learning process. The objective metrics and subjective userstudy reveal that the proposed models can begin to learn aboutmusic. Although the musically and aesthetically generated tracks do not match the level of human musicians yet, the proposed model has certain desirable properties and we hope that future research will be able to further improve it.

References

1. *Boulanger-Lewandowski, N., Bengio, Y., & Vincent, P. (2012). Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription. arXiv preprint arXiv:1206.6392.*
2. *Briot, J. P., Hadjeres, G., &Pachet, F. D. (2017). Deep learning techniques for music generation — a survey. arXiv preprint arXiv:1709.01620.*
3. *Choi, K., Fazekas, G., & Sandler, M. (2016). Text-based LSTM networks for automatic music composition. arXiv preprint arXiv:1604.05358.*
4. *Dong, H. W., Hsiao, W. Y., Yang, L. C., & Yang, Y. H. (2017). MuseGAN: Demonstration of a convolutional GAN based model for generating multi-track piano-rolls. ISMIR Late Breaking/Demos.*
5. *Eck, D., &Schmidhuber, J. (2002). A first look at music composition using lstm recurrent neural networks. IstitutoDalleMolle Di StudiSullIntelligenzaArtificiale, 103, 48.*
6. *Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... &Bengio, Y. (2014). Generative adversarial nets. In Advances in neural information processing systems (pp. 2672–2680).*
7. *Hadjeres, G., Pachet, F., & Nielsen, F. (2017, July). Deepbach: a steerable model for bach chorales generation. In International Conference on Machine Learning (pp. 1362–1371). PMLR.*
8. *Hochreiter, S., &Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735–1780.*
9. *Moorer, J. A. (1972). Music and computer composition. Communications of the ACM, 15 (2), 104–113. 47*
10. *A. Nefian and M. Hayes, " An embedded hmm-based approach for face detection and recognition," In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 6, pp. 3553–3556, 1999.*
11. *U.S. Department of Defense, "Facial Recognition Vendor Test, 2000," Available: http://www.dodcounterdrug.com/facialrecognition/FRVT2000/frvt2000.htm*
12. *Banu, Danciu, Boboc, Moga, Balan; "A novel approach for face expression recognition", IEEE 10th Jubilee International Symposium on Intelligent Systems and Informatics 2012.*
13. *Wang Zhen, Ying Zilu; "Facial expression recognition based on adaptive local binary pattern and sparse representation", 2012 IEEE.*
14. *Deepti, Archana, Dr. Jagathy; "Facial expression recognition using ANN", IOSR Journal of Computer Engineering 2013. Special Conference Issue: National Conference on Cloud Computing & Big Data 162.*
15. *Jizheng, Xia, Lijang, Yuli, Angelo; "Facial expression recognition considering differences in facial structure and texture", IET Computer Vision 2013.*
16. *Jiawei, Congting, Hongyun, Zilu; "Facial expression recognition based on completed local binary pattern and sparse representation", Ninth International Conference on Natural Computation (ICNC) 2013.*