Curbed Music Generation With Deep Learning

Abde-Abitalib Merchant[1]

[1]Bachelor in Computer Engineering, Fr. Conceicao Rodrigues College of Engineering, Mumbai

***Abstract*—**Artificial intelligence solutions have been brought into the creative field such as generations of art and music. Generating music via deep learning is a challenging problem, however recent developments such as openAI's MuseNet, CTRL (Creator Technology Research Lab) at Spotify, and Google's Magenta have shown us that there can be viable solutions. However, these technologies use very complex models trained over thousands of samples of data. This paper presents a survey on different models used in the past and also proposes a new model which uses LSTMs (Long Short Term Memory) and a small dataset to show that a relatively basic model is also functional in producing a sequence of notes which is capable of demonstrating properties resembling a nice sounding piece of music. We also show that a model produces satisfactory results when presented with a small dataset to train. This paper also discusses the challenges that accompany and discusses the strategy, and architecture that is involved in creating a system that can successfully generate a piece of music rather than a sequence of notes.

***Index Terms*—**Music, LSTM, Melodies, MIDI, Deep Learning,Machine Learning

elements are sufficient to generate satisfactory results, with the added benefit of being quantifiable. Those terms are listed as follows:

- Pitch - pitch, in music, is defined as a position of a single sound in the complete range of sound. Sounds are higher or lower in pitch according to the sound waves' frequencyof vibration[5].
- Melody - Melody is the LINEAR/HORIZONTAL presen- tation of pitch (the word used to describe the highness or lowness of a musical sound).[6]
- Harmony - Harmony is the VERTICALIZATION of the pitch.[6].
- Chords - Often, harmony is thought of as the art of combining pitches into chords (several notes played si- multaneously as a "block")[6].
- Chord Progressions – A succession of chords is definedas a chord progression.

I. INTRODUCTION

USIC generation is a challenging problem in whichmany advancements have been made by training deep learning models over a large dataset. However, it is not assuredthat the quality of music generated will be human-like. Thispaper discusses the effects of using a smaller dataset. Ourmodel also emphasizes constraints like Structure(in contrastto wandering music without a sense of direction) and Creativ-ity(in contrast to imitation and risk of plagiarism) suggested by JP Briot and Francois Pachet[1].

A. Music and its related terminology

Epperson defines music as 'art concerned with combining vocal or instrumental sounds for beauty of form or emotional expression, usually according to cultural standards of rhythm, melody, and, in most Western music, harmony.' [2]. Merriam- Webster defines music as 'the science or art of ordering tones or sounds in succession, in combination, and in temporal relationships to produce a composition having unity and continuity'[3]. Music theory, the study of the

structure and patterns of sounds produced, comprises a collection of rules that outlines how pieces are fitted together [4]. Although music and music theory consists of many elements that are important and work together, such as key signature, time signature, pitch, scales, modes, consonance, dissonance, etc, however, we are concerned with only a few terms that are necessary to understand in order to generate music. This is because these

B. *Historical review of music generation*

The generation of music can be attributed, but not au- thenticated all the way back to the classical era of music. German musical dice game (or 'Musikalisches Würfelspiel'), supposedly invented by Mozart, uses random numbers to select the pre-composed music pieces in a particular sequence, and the combination of those pieces leads to a complete piece of music. This is a small example of algorithmic composition, which is defined as the study of the combination of two separate fields: computer science and music composition to create systems that are capable of generating music. There have been many different approaches to generating music such as the Genetic approach or the Markov approach. Lejaren Hiller and Leonard Isaacson developed The Iliac Suite for String Quartet, which is the first ever piece of music that is said to be composed by a computer [7]. Statistical models have also been used to generate music wherein an HMM approach used by Allan was utilized in chorale melody harmonization [8]. With the increase in popularity of Deep Learning, it became easier to train systems from a large dataset. There was also more control over the exact type of music that you'd want to generate (such as blues, country, classical, etc).

C. *Different modern techniques for music generation*

As more advancements have been made in deep learning techniques and hardware, it has made it possible to advance deep learning models, which learn from a huge dataset to perform complex tasks such as image recognition, text syn- thesis, etc more efficiently. In the case of music generation, advancements in deep learning have led to some remarkable results done previously. With the goal of autonomy in mind we can move further with the different kinds of systems that have been created with the purpose of automating the process of music composition.

Different researchers have proposed solutions using various learning algorithms. They are mentioned below:

Mangal et al use an LSTM model in which the methodology includes a single LSTM which is used to predict the time at which a note is played and the sequence of notes. The model learns a sequence of notes over a single-layered LSTM network[9].

A bi-axial LSTM network architecture is used to learn polyphonic music in the alignment of musical rules. It is trained with a convolutional kernel[10].

Huang et al also use a 2-layer LSTM with the goal of end-to- end using a deep neural network, i.e an LSTM RNN(Recurrent Neural Network), with a goal of composing music that has both harmony and melody[11].

Jiang et al have also devised a way to generate music usinga bidirectional LSTM model, which had a higher performancethan the unidirectional bi-axial model. This model can learnthe intricate relationship between notes and predict the nextnotes via probability from the time and pitch dimensions[12].

With networks such as GANs, Transformers, etc, it has been easy to generate music that is layered and also fast. MIDINet, a convolutional Generative Adversarial Network [13], followsthe basic principle of a GAN described by Goodfellow et al [14] in which 2 models are trained simultaneously, a generator G and discriminator D, which are adversarial in nature. This model can generate melodies of up to 8 bars either from scratch or an already existing bar of music.

In a similar vein, MuseGAN consists of 3 models: the jam- ming model, composer, and hybrid models. These 3 models assist in multi-layered music generation and can generate 4 bars of music without any human input. In addition, it also has a human-AI cooperative model based on the MuseGAN model: given a specific track by a human, it can generate 4 additional compatible tracks[15].

DeepJ, an end-to-end generative model is a model which has the ability to generate music based on a

specific style of a composer. Its novelty lies in the fact that it allows the user to generate music with tunable parameters, being an improvement over the biaxial LSTM approach[16].

Mittal et al have discovered a way to train diffusion models to generate music by using a Denoising Diffusion Probabilistic Model (DDPM), which is mainly used in the domains of audio and image, rather than symbolic music generation. This method is non-autoregressive and learns to generate sequences of the latent embeddings and can generate music parallelly[17].

Most of the models mentioned use MIDI files as data, but Sander Dieleman et al have proposed a system that is modeled over raw audio and generates stylistically consistent music which contains nuances and subtleties which lack when MIDI files are used. An autoregressive discrete autoencoder model is able to capture the long-range correlation between music.[18]. There is one thing common for all the above methods mentioned: the usage of a large dataset. In an analytical model, a large dataset is good, because more data equals more information which can be beneficial. However, when considering a generative problem in a creative field(such as art or music), can the same results be obtained on a smaller dataset? This paper discusses the effects of using a smaller dataset. One thing to consider is our goal: to achieve autonomy
i.e. we want our system to generate music independently. In this paper, we use an LSTM with a small dataset.

II. METHODOLOGY

A. *Data*

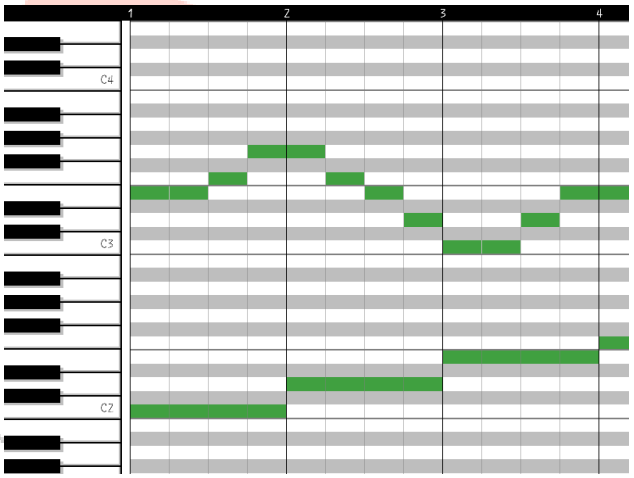
The representation of data chosen here is a MIDI file. Mu- sical Instrument Digital Interface (MIDI) is a file format that contains electronic information about music representation. It does not possess the actual sound wave used, rather it conveys information about what notes are to be played to a midi- compatible instrument. (Refer Figure 1)

Fig. 1. One of the way to visualize a MIDI file: A piano roll, which shows the placement of each note at a particular time period and the frequency of the note simultaneously

B. *Dataset used*

The dataset used for the training of the model is Lo-Fi Hip Hop MIDIs Dataset by Zachary Katsnelson. It consists of 93 midi files, all of which possess similar sizes. The difference arises in the key, or tonic of the files, which consists of major chords and minor chords as well. The different keys are important for the model to learn different chord patterns, notes, and progressions. The model learns these chords and emulates them, and the chords are grouped together with the help of a module known as music21, created by Michael ScottCuthbert [19].

C. *Model Creation and Architecture*



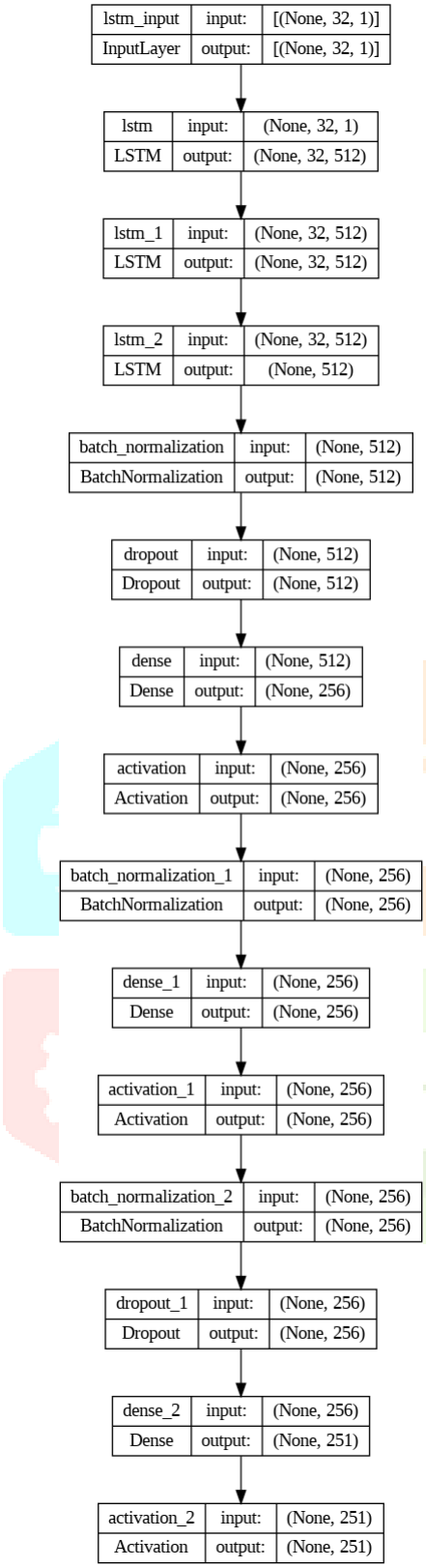


Fig. 2. The above figure describes the model architecture of the LSTM model

From the dataset, we create an input of 32 sequences, i.e 32 notes and their corresponding outputs. The sequences are prepared from all the notes and chords from each midi file in the dataset. We use the music21 module to parse notes in the file by different instruments, notes, and chords. The final layer is a softmax layer which gives us the next best probable note in the sequence. We utilize some regularization techniques such as recurrent dropout for the LSTM layers, and regular dropout. Standard techniques such as Batch Norm are also utilized. Refer to Figure 2 for a detailed view of the flow of data and model architecture.

D. Model Performance

To understand whether our model has performed well and can generate the next best notes for unseen data, a validation set was created. We then measure the accuracy and the loss and visualize it via a graph.

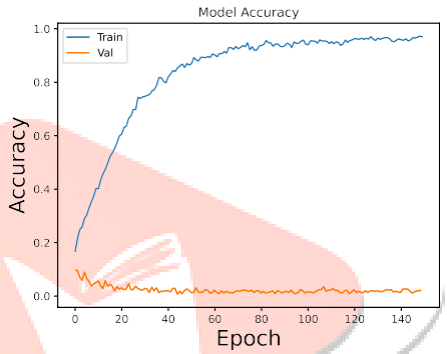


Fig. 3. The above figure describes the accuracy of the model for the train and validation sets

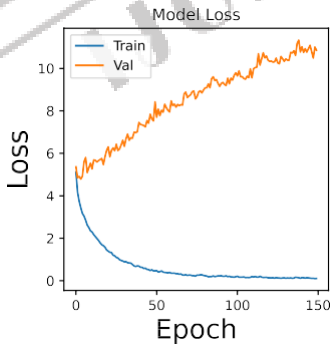


Fig. 4. The above figure describes the loss of the training and validation sets.

From this, we can notice a massive difference between the accuracy of the model on the training set as well as the validation set (Figure 3). The loss plot also tells a similar story, with a massive difference between the loss on the training set and the validation set. (Figure 4) We can see that the model performs poorly on unknown data, but since we are

training the model on major progressions and the majority of the chords, and since we also give random notes to the model at the beginning for generation, it does not reflect in the final generation. When the validation set is discarded, and the model is trained on the entire dataset, there is a slight increase in performance (Figure 5).

The number of notes that the model can generate after getting an input of random notes can also be controlled. A

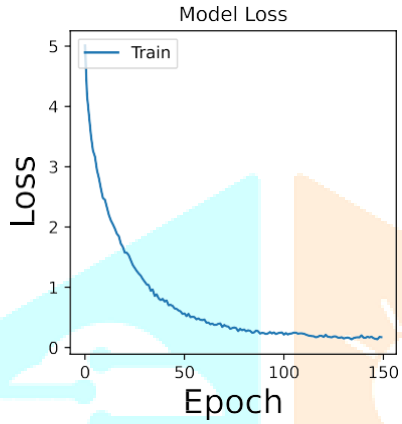


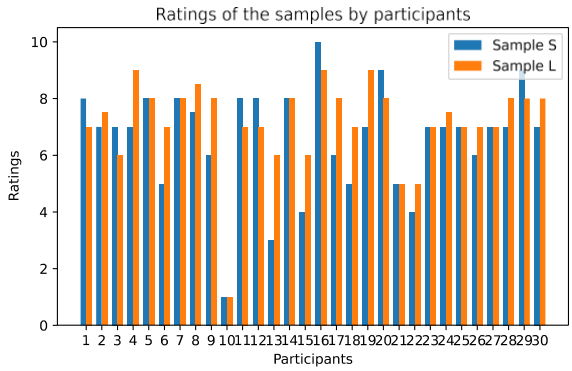
Fig. 5. The above figure describes the loss plot for the model trained on the entire dataset

short version length of around 10 seconds, a medium version of length 50 seconds (Sample M), and a long version of 130 (Sample L) seconds were all generated from different input notes, as a way to test the capabilities of the model in different time frames. From these, the medium and long snippets were used for determining the musical aesthetic, since the short model was not long-lasting enough to convey the capabilities of the learned model. On blind listen, Sample M, contained structure, patterns, and progression along with the melody, satisfying the criteria of Structure and Creativity. Similarly, Sample L started out well but eventually began to have little progression and structure and more of a random guess of notes of the model, thus only partially satisfying the criteria of control and structure.

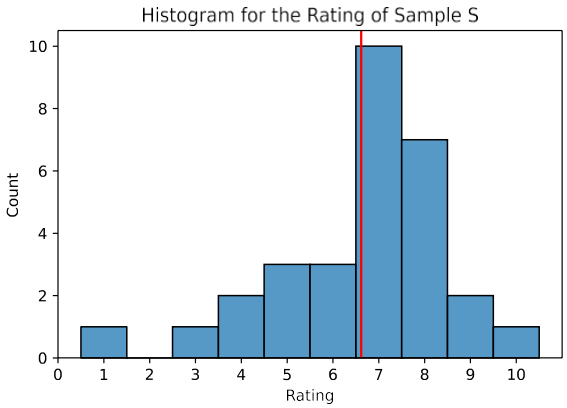
III. RESULTS

It is difficult to determine objectively the aesthetic value of the music generated. So, to test the aesthetic quality of the music generated, the testing method by Huang et al [11] was adopted here as well. A blind test via a survey was given to a number of people, where they were asked to rate the musical samples between 1 and 10, where 1 = random noise, 5 = musically plausible, and 10 = music composed by a composer. The 2 samples given for rating were the ones generated by the small dataset-trained model (Sample S) and a similar model trained on a large dataset. Participants were asked to rate the sample and the results were mixed. From the results shown in Figure 6, the average rating for Sample S is 6.61 (Figure 6b), and the rating for Sample L is 7.15 (Figure 6c).

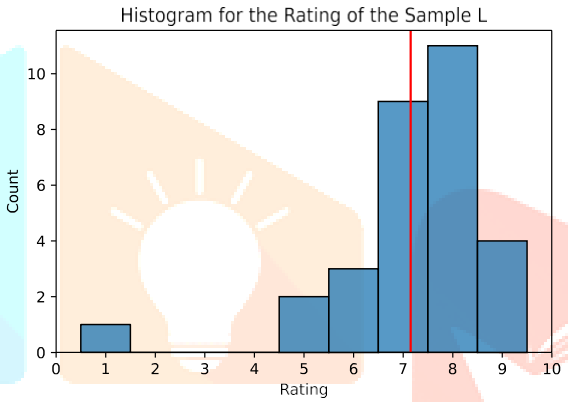
From the process, we can see that a small dataset used can result in music that can be deemed ‘musically plausible’. This also showcases the fact that with a complex model and utilizing an architecture, we can obtain satisfactory results, in addition to the norm which says for ’creative’ tasks, to get good results, we require a lot of data. We can also argue the fact that using a small dataset might lead to a model which can generate music that is ‘imperfect’, which in this scenario can be a good thing because studies by Hamilton et al in the book ’The Aesthetics of Imperfection in Music and the Arts’ says that imperfection has its own aesthetic, thereby stating that the quality of imperfection in music increase the aesthetic and enjoyment of music [20].



(a) Ratings of the samples



(b) Histogram for Sample S



(c) Histogram for Sample L

Fig. 6. The above figure is the results of the survey conducted

IV. CONCLUSION

This paper displays that a small dataset when paired with the right architecture of model and techniques, can gener- ate music without any human assistance. We can conclude that a Deep Recurrent Neural Network when trained off a small dataset, is capable of generating music of sufficient length, displaying properties of chords, and progression, and satisfying the criteria of creativity and structure. A complex model using LSTMs results in a sound that is aesthetically pleasing or musically plausible. We believe that improvement is definitely possible, whether it be different combinations of notes, hyperparameters, and other variables which can improve the performance of the model. With this paper, we have obtained a baseline model which when trained with a limited dataset, provides satisfactory results and can be used as a baseline model for comparison with even more complex and

ACKNOWLEDGMENT

This research was supported by Fr. Conceicao Rodrigues College of Engineering. The author would like to thank Dr. Sunil Surve, Professor for their criticism that shaped and greatly improved the manuscript. The author would also like to thank the participants of the survey.

REFERENCES

[1] J.-P. Briot and F. Pachet, "Music generation by deep learning-challenges and directions," *arXiv preprint arXiv:1712.04371*, 2017.

[2] Gordon, Epperson, "The definition of music,"https://www.britannica.com/art/music (Aug. 2

[3] Merriam Webster, "The definition and meaning of music,"

upcoming models in the future.
https://www.merriam-webster.com/dictionary/musi (n.d).

[4] Scarlett Helfer, "Music theory," December 5th 2022, available at: https://study.com/learn/lesson/music-theory-overview-concepts-history.html (Dec. 2022).

[5] Brittanica, "The meaning of pitch," https://www.britannica.com/art/pitch-music (Jun. 2019).

[6] Western Michigan University, "The elements of music," https://wmich.edu/mus-gened/mus150/Ch1-elements.pdf (2001).

[7] L. M. Isaacson, *Experimental Music; Composition with an Electronic Computer by Lejaren A. Hiller, Jr. and Leonard M. Isaacson*. New York, McGraw-Hill, 1959.

[8] D. Conklin, "Music generation from statistical models," in *Proceedings of the AISB 2003 Symposium on Artificial Intelligence and Creativity in the Arts and Sciences*. Citeseer, 2003, pp. 30–35.

[9] S. Mangal, R. Modak, and P. Joshi, "Lstm based music generation system," *arXiv preprint arXiv:1908.01080*,2019.

[10] N. Kotecha and P. Young, "Generating music using an lstm network," *arXiv preprint arXiv:1804.07300*, 2018.

[11] A. Huang and R. Wu, "Deep learning for music," *arXiv preprint arXiv:1606.04930*, 2016.

[12] T. Jiang, Q. Xiao, and X. Yin, "Music generation using bidirectional recurrent network," in *2019 IEEE 2nd International Conference on Electronics Technology (ICET)*. IEEE, 2019, pp. 564–569.

[13] L.-C. Yang, S.-Y. Chou, and Y.-H. Yang, "Midinet: A convolutional generative adversarial network for symbolic-domain music generation," *arXiv preprintarXiv:1703.10847*, 2017.

[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[15] H.-W. Dong, W.-Y. Hsiao, L.-C. Yang, and Y.-H. Yang, "Musegan: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[16] H. H. Mao, T. Shin, and G. Cottrell, "Deepj: Style- specific music generation," in *2018 IEEE 12th Inter- national Conference on Semantic Computing (ICSC)*. IEEE, 2018, pp. 377–382.

[17] G. Mittal, J. Engel, C. Hawthorne, and I. Simon, "Sym- bolic music generation with diffusion models," *arXiv preprint arXiv:2103.16091*, 2021.

[18] S. Dieleman, A. van den Oord, and K. Simonyan, "The challenge of realistic music generation: modelling raw audio at scale," *Advances in Neural Information Process-ing Systems*, vol. 31, 2018.

[19] M. S. Cuthbert and C. Ariza, "music21: A toolkit for computer-aided musicology and symbolic music data," 2010.

[20] A. Hamilton and L. Pearson, *The aesthetics of imperfec- tion in music and the arts: Spontaneity, flaws and the unfinished*. Bloomsbury Publishing, 2020.