2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES) | 978-1-6654-3521-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICSES52305.2021.9633906

Music Generation for Novices Using Recurrent Neural Network (RNN)

Sahreen Sajad, S Dharshika and Merin Meleet

Department of ISE, R.V. College of Engineering, Karnataka-India,

Abstract :- Listening to music is a pastime most people enjoy. We're all fascinated with music and resort to listening to it in times when we're in a good mood and also while in distress. While a variety of applications and softwares exist that let musicians make music, there is not much development in the field for novices who do not understand music. This paper aims to change that. Not everyone should need to be an expert in the field to be able to create melodious pieces of music. This paper gives an approach to be able to do the same using Recurrent Neural Networks. The idea is to build a model that trains using existing melodies or instrumentals and generate new music based on the training. The approach will not only be helpful to people who do not know the field well but also to musicians to be able to generate fine quality music that can be developed further to make decent length songs. We aim to create music without having a need to play musical instruments physically.

Keywords—Music Generation, Recurrent Neural Networks (RNN), chord, note, MIDI, LSTM, Sheet Music

I. INTRODUCTION

The music industry is booming day-by-day. More and more artists create new music but the effort that goes into creating these melodies is massive. The process can be made easy and less time consuming if music generation is automated. Automated music can not only let people who have no knowledge about music produce music, but also let musicians to create melodies that can be further developed to make full-fledged soundtracks. The domain is a field of research for many years now.

Generating music using AI is no more a novelty. There are a plenty of methods to generate music using AI services. The software that generates music takes tons of music material which consist of music chords, tones, sequence etc., and it studies the pattern to generate its own music sequence. There are some platforms that help in generating music such as Amper, IBM etc., they give the output in MIDI format. Amper has an interface that is easy to use and understand and it gives the output in audio format.

Automated music began with Chinese wind chimes. A Japanese garden music device called Suikinkutsu also produced automated music. It has an upside-down pot having a hole. Previously, generative grammars were used to automate music but this has been replaced with machine learning and deep learning. MIDI is another protocol that is being used. It is used for a variety of electronic musical instruments for recording and editing soundtracks.

Here, some existing music melodies will be used to train a model. The model is supposed to understand sequences and patterns and be able to generate music based on these patterns. The MIDI files will be used as an input to the Recurrent Neural Network (RNN) and generate new patterns of melodious music. RNN is a form of neural networks that does not take a single input or produce a single output but it takes a series of inputs and produces a series of outputs. A number of hidden layers are present that take the input and produce the corresponding output in the next hidden layer and so forth until the final output is produced.

Though, a detailed study or knowledge of music is not required to use this model. But a basic knowledge on music theory can surely make the task easier.

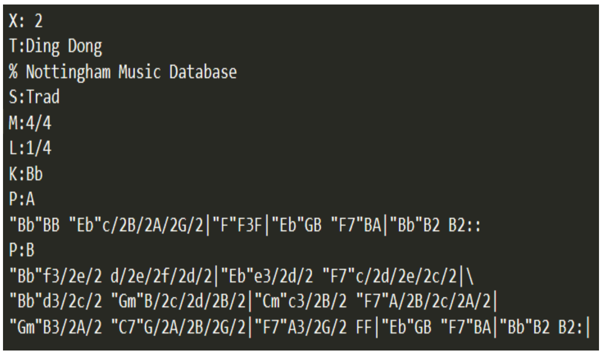
ABC Notation

There are seven notes in music [1]. These are represented with the letters A, B, C, D, E, F, G in musical notation for computers. Pitch refers to the lowness or highness of sound. A set of notes exist in an octave. What octave a note belongs to determines the pitch of the sound. For example, G1 which is a G note in octave 1 will be much deeper than G7, a G note in octave 7.

A key determines the majority of notes that form a basis for music. Scales are then built notes around a specific key. Songs which sound happy are mostly based on C major scales and sad music is generally based on D or E minor scales. A group of notes form a chord. For example, the A major chord is made of A, C sharp and an E. A-G is used to refer to major chords and a-g is used to refer to minor chords.

Besides the letters, some other notations are also used to refer to "flat" or "sharp" chords. The filename extension is .abc. In the initial part of the notation, letters are followed by colons. X: is used to refer to the number of tunes in a file, T: is the title, M: is the time signature, L: is the length of the note by default, R: is the type of tune and K: is the key.

Various tools exist that let you convert the music in ABC notation to traditional music notation, for example, Score extension. It renders music and transforms them into audio files.



```
X: 2
T:Ding Dong
% Nottingham Music Database
S:Trad
M:4/4
L:1/4
K:Bb
P:A
"Bb"BB "Eb"c/2B/2A/2G/2|"F"F3F|"Eb"GB "F7"BA|"Bb"B2 B2::
P:B
"Bb"f3/2e/2 d/2e/2f/2d/2|"Eb"e3/2d/2 "F7"c/2d/2e/2c/2|\
"Bb"d3/2c/2 "Gm"B/2c/2d/2B/2|"Cm"c3/2B/2 "F7"A/2B/2c/2A/2|
"Gm"B3/2A/2 "C7"G/2A/2B/2G/2|"F7"A3/2G/2 FF|"Eb"GB "F7"BA|"Bb"B2 B2:|
```

Figure 1. Music Representation in ABC Notation

A. *Sheet Music*

Sheet Music [2] is a printed form or handwritten form of music notations. No requirement of sheet music is needed for the implementation. But learning sheet music could help gain access to a greater number of tunes that are not yet available in ABC notation.

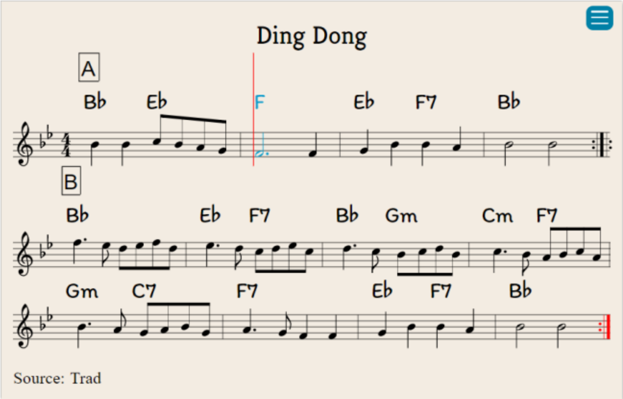


Figure 2. Sheet Music

B. *Musical Instrument Digital Interface (MIDI)*

Recording and editing of music is done through a protocol called Musical Instrument Digital Interface (MIDI). Maintained by MIDI Manufacturers Association (MMA), an association called The MIDI Association (TMI) is also established for the general public for creating and working with MIDI.

MIDI is not music in an audio format, but a set of instructions which can be further translated to actual sounds. A number of softwares allow this to happen. These include Fruity Loops (FL) Studio and Synthesia.

Translation of MIDI into sheet music is quite easy. This lets musicians to be able to write music without the need of having or playing musical instruments physically.

MIDI messages are of two types; system messages and channel messages. Our implementation is only concerned with the channel messages of MIDI.

II.RELATED WORK

Earliest papers revolving around the domain have not taken into consideration a global structure in the music. This means that the music generated has no harmony as such. Harmony is nothing but a proper arrangement in the notes. A proper rhythm and a better arrangement can be introduced which takes into consideration long chords and rests in between. The output though may not be a decent length song with proper distinction between various parts of a song, but will be a melody which will be passable as music to the human ears.

Authors in [3] had the objectives to generate music with a given sentiment using deep learning. There was a gap identified that it also generated ambiguous negative pieces. The result was obtained using a generative mLSTM that can be controlled to generate symbolic music keep in mind the objective i.e., to use a given sentiment. The mLSTM was controlled by optimizing the model weights of specific neurons that are responsible for the sentiment analysis. The model obtained a good prediction accuracy.

Authors in [4] stated that the model studies the influence of expressiveness in music and it doesn't improve the musical compositions. The objective was to give the input to LSTM and dropouts in layers and receive the output in the MIDI format. The perception of an artificial music composition was the closer to one that we get in a live performance.

Authors in [5] gave a novelty melody composition method that strengthened the original Generative Adversarial Network (GAN) model based on individual bars. The experimental results showed that music generated had more similarity with the real music structure with a minute difference of 8% than that of traditional music structure.

Author in [6] had the objective to evaluate and verify music generation with different networks structures. Some of them like LSTM, RNN and GRU (Gated Recurrent Units). The overall training process was ideal because all the proposed network structures reached up to 80% accuracy in the experiment.

Authors in [7] proposed an algorithmic melody generation using a Generative Adversarial Network without recurrent components. There was a gap identified that an abundance of Irish music exists only in ABC Notation and the result obtained were graphs that were normed to a min value of 3.8 and max value of 6.

Authors in [8] aimed to generate music using a self-correcting, non-chronological, autoregressive model. The drawback that the data representation doesn't convey important characteristics like note velocity, repeated noted or the duration of note. These can lead to technical and emotive quality of music. This work's results showed that a thorough quantitative metrics and human survey evaluation that the approach gives better results than order less NADE and Gibbs sampling approaches that were shown in the paper.

Authors in [9] proposed a system that can generate plausible music for a video clip that has no audio. There was a drawback that the system couldn't generate waveform from MIDI events. The performance of this algorithm was better than baselines. There were correlations between visual and musical signals that were set through body key points and MIDI representations.

III.METHODOLOGY

Character RNN Model

Music is not a single input that should produce a single output. Rather, it is a sequence or series of inputs. As a result, Character-Recurrent Neural Networks can be used to process the input into hidden layers and obtain the consecutive outputs to obtain a final output. Here, a character-level RNN will be used to input music as a series of characters and find the prediction of various hidden layers that give the final output. Every input of character will give an output, so the no. of inputs given is equal to the number of outputs received. Thus, many-to-many RNN can be used. Every unit of this many-to-many RNN will be a repeating structure. For every input 'a' given to the model, the model will be expected to generate an output 'b'. The output 'b' will then be given as an input and some output 'c' will be expected. Now, output 'c' will be given as an input and output 'a' will be expected. Similarly, more sequences will be generated. This process has to be repeated until all our inputs are sent into the model[10].

Categorical Cross-Entropy loss function will be used which helps to calculate the loss based on expected class and actual class. This loss helps to generate the probability based on the distance between the expected class and actual class.

The last layers of the RNN model will use the "SoftMax Activations". The number of these units will depend on the number of unique characters which will be used in the training model. Back-propagation will be used and continuous iteration will be done until a convergence is met. Adam optimizer will be used for handling the sparse gradients[11]. The result will be a trained model that will have the ability to take music notes as inputs and learn the sequences to generate new music. While using the model, any of the characters that were fed during the training phase will be used as an input, automated characters will be sent by the RNN model as an output based on what it's learnt during the training phase.

There are 3839 characters in the dataset, of which 73 unique characters. Hence, there will be a total of 3 batches. The size of the batch is 16 and the sequence length is 63 hence a total of 3 batches are created.

IV.DATA BATCHES AND RNN UNITS

The data is fed into the model in the form of batches. The batches can be specified using the size and length of sequences. Every character used for training is indexed with a particular value[12]. A key also exists which has an index. The value of the index is the character itself.

In the first RNN input, zero input is given which is a sort of dummy input[13]. This is because in RNN, previous outputs are treated as inputs to gain corresponding outputs. Since, in the beginning, there is no output at all, zero is given as an input in the first iteration. Then, the first output and the next character are fed together as the input in the second iteration. This is how many-to-many RNN works. The same procedure follows in the next iterations.

Consider 256 RNN-LSTM. LSTM or Long Short-Term Memory has feedback connections. RNNs store data of only the previous state. In automated music generation, this cannot be very helpful. This is where LSTM comes into the picture[14]. Additional memory cells are added due to LSTM that have three kinds of gates. These are input, output, and forget. Input gate is responsible for data fed into memory, output gate is responsible for data which is to be sent to the next layer, and forget is responsible for loss in the memory. It is because of LSTM, that data is not present of the previous state only[15]. In each iteration, all RNN units will generate an output which will be fed as an input to the next layer as well as the same output will become the input to the same RNN unit. A "return sequence" parameter is used which is set to False b default. Turning it to True will generate output for every single character. After these particular layers, "Time Distributed" layers will be used. These will have activations of "SoftMax" as well[16].

Another parameter used would be "stateful". If stateful is set as true then every sample at some index will be used as initial state for sample of that index in the next batch.

V. DATA VISUALIZATION

The dataset consists of 13 different Christmas tunes in the for of abc notation. The model has been trained 4 times to achieve the accuracy over 90% for the music to be melodious[17]. The change in accuracy and loss is visualized after every training the model each time.

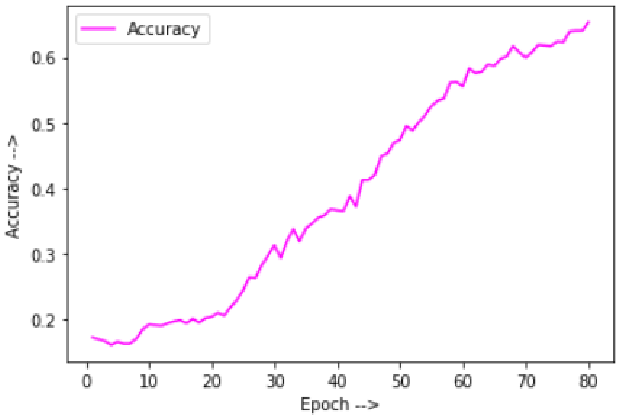


Figure 3: Accuracy v/s No. of Epochs (Train 1)

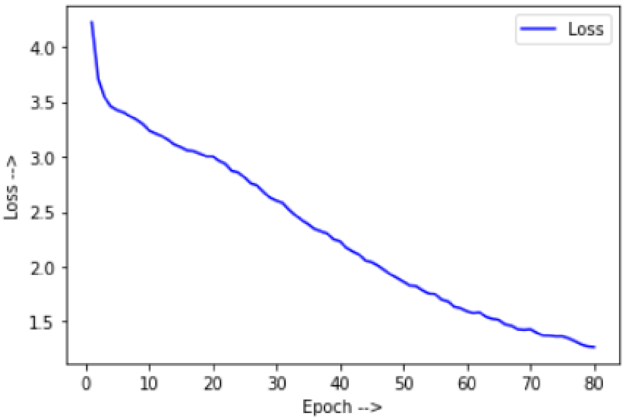


Figure 4: Loss v/s No. of Epochs (Train 1)

After training the model once, the accuracy of 65% was achieved and the loss also plummeted after 80 epochs. To improve the accuracy, the model weights from the previous training are loaded again with some more extra LSTM layers with more LSTM units[18].

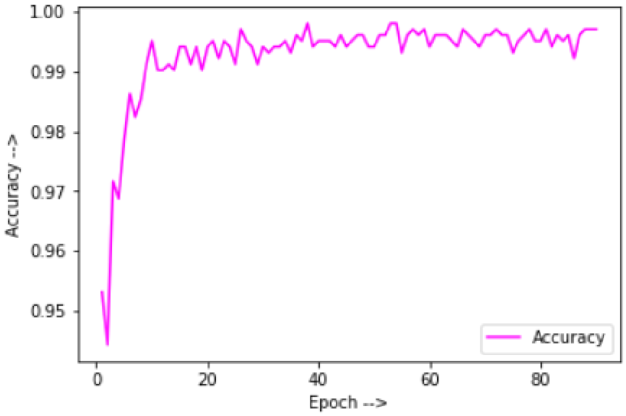


Figure 5: Accuracy v/s No. of Epochs (Train 4)

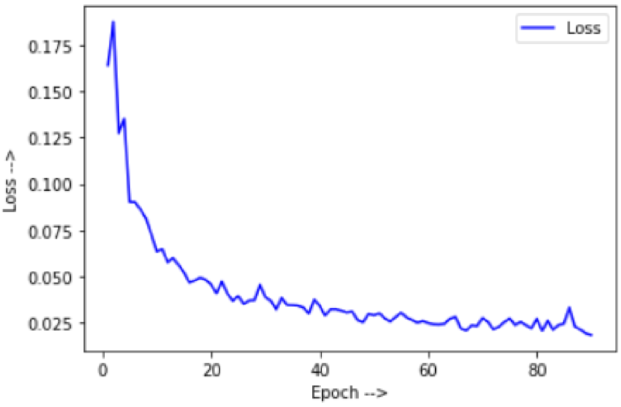


Figure 6: Loss v/s No. of Epochs (Train 4)

After adding some more layers, finally after training the model four times, the accuracy of 99% was achieved and loss was also very less compared to the first training.

VI.RESULTS

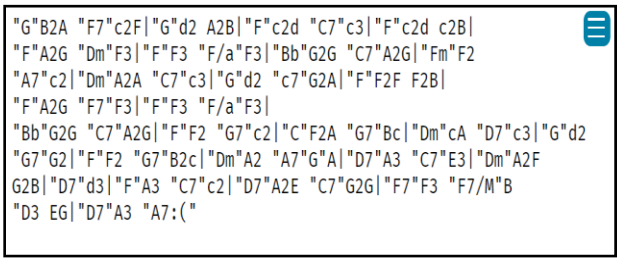
A. In the form of MIDI

After training the model and finding the best weights, prediction is to be made. Any of the dimensional output at each time stamp can be given as an input. The same number of probability values will be generated as the number of dimensional outputs using the SoftMax layer. The next character will be chosen among the returned probability values[19]. This choice is made probabilistically and not deterministically. What this means the choice is made based on likelihood of a certain outcome and not on information that we're certain is true. The chosen character s then treated as input again into the model and some length of music will be generated by continuous concatenation of the character outputs.

B. Conversion of MIDI into sound

Chrome driver is an executable that is used to control chrome. It is an open-source tool for automatically using the webapps across browsers like chrome, Firefox etc. "abcjs" online tool is accessed to convert the music to midi file. Using the music can also be played along with the sheet music displayed on the side[20].

For generating the music (Christmas Tune): input character was given as 67, 90 was the epoch number model loaded and length of the sequence was given as 450. The output is as follows:



```
"G"B2A "F7"c2F|"G"d2 A2B|"F"c2d "C7"c3|"F"c2d c2B|
"F"A2G "Dm"F3|"F"F3 "F/a"F3|"Bb"G2G "C7"A2G|"Fm"F2
"A7"c2|"Dm"A2A "C7"c3|"G"d2 "c7"G2A|"F"F2F F2B|
"F"A2G "F7"F3|"F"F3 "F/a"F3|
"Bb"G2G "C7"A2G|"F"F2 "G7"c2|"C"F2A "G7"Bc|"Dm"cA "D7"c3|"G"d2
"G7"G2|"F"F2 "G7"B2c|"Dm"A2 "A7"G"A|"D7"A3 "C7"E3|"Dm"A2F
G2B|"D7"d3|"F"A3 "C7"c2|"D7"A2E "C7"G2G|"F7"F3 "F7/M"B
"D3 EG|"D7"A3 "A7:("
```

Figure 7: Output in abc notation

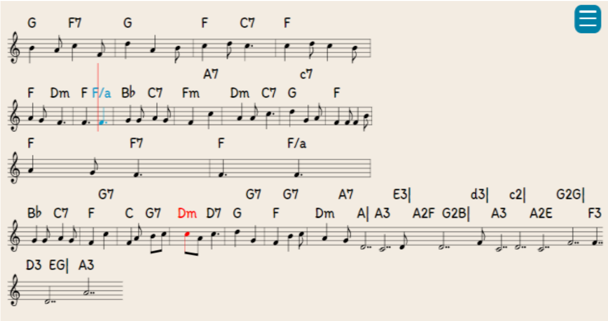

G F7 G F C7 F
A7 c7
F Dm F F/a Bb C7 Fm Dm C7 G F
F F7 F F/a
G7 G7 G7 A7 E3| d3| c2| G2G|
Bb C7 F C G7 Dm D7 G F Dm A| A3 A2F G2B| A3 A2E F3
D3 EG| A3

Figure 8: Output in sheet music [21]

Superfluous information in the MIDI files is removed including velocity and time. The remaining information is then used to convert to note objects. Encoding is then done which helps to feed this data inside the network that helps to listen to the music[22]. A number of softwares and applications also exist that let you listen to the audio directly by just copying the output of the RNN model.

MIDI encoding process results in a matrix. The columns of the matrix depict the time and the rows depict the pitch.

Decoding of the matrix can also be done to receive the corresponding MIDI files[23].

VII.SCOPE AND FUTURE ENHANCEMENTS

The automated music generated is of decent quality and length, it is not a full-fledged piece of melodious track. The various parts in a song including the chorus, verses or bridge and their respective changes in chords and scales is not taken into consideration yet. More tunes, instrumentals, and pieces of music can be added to train the model well.

Besides, unknown notes need to be filtered. These can be handled in the future enhancements or replaced with known notes in the existing data. The length of the music output needs to be adjusted as well as the present output is just a short melody.

VIII. CONCLUSION

This paper proposes a method to generate new automated music for novices using RNN.

As music is a hobby that most people enjoy, this paper aims to help build a model that can let beginners as well as musicians to create new melodies without having a deep knowledge about the domain. All that is required is a set of existing tunes or instrumentals that will be fed into the model. The model is expected to train using these music patterns and generate output by learning what patterns are pleasing to hear. The model does not copy or imitate the existing dataset but learns the kind of notes and chords that together sound melodious and replicate the process to create music of some short length.

This goal of this work is not only help novices to be gifted with the ability of creating music but also serve as a basis for musicians to create songs using these short melodies. We aim to automate, simplify and fasten the process of making music without having to use physical instruments or having a knowledge on how to play these instruments. All can be conveniently done with the click of a mouse and the press of a button.

REFERENCES

[1] Douglas Eck and Jurgen Schmidhuber, "A first look at music composition using lstm recurrent neural networks. No. IDSIA-07-02, 2002.

[2] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space.

[3] Ferreira, Lucas N. ,Jim Whitehead, "Learning to generate music using sentiment", March 2019

[4] Penousal Machado, Ana Cláudia Rodrigues and José Maria Simões, "Deep Learning for expressive Music Generation", ARTECH 2019: 9th International Conference on Digital and Interactive Arts, October 2019

[5] Sejun Jang, Yunsick Sung and Shuyu Li "Automatic melody generation using enhanced GAN", DOI: 10.3390/math7100883, September 2019

[6] Jiatong Xie, "A Novel Method of Music Generation Based on Three Different Recurrent Neural Networks", DOI: 10.1088/1742-6596/1549/4/042034, June 2020.

[7] Mitchell Billard, , Antonina Kolokolova and Moustafa Elsisy, " GANs & Reels: Creating Irish Music using a Generative Adversarial Network"

[8] Wayne Chi and Rahul Suresh, "Generating Music with a Self-Correcting Non-Chronological Autoregressive Model", August 2020

[9] Chuang Gan, Antonio Torralba, "Foley Music: Learning to generate Music from videos", July 2020

[10] Sutskever, I., Vinyals, O. and Zaremba W." Recurrent neural network regularization", arXiv preprint arXiv:1409.2329 (2014)

[11] G. Fazekas, K. Choi and M. Sandler, "Text-based LSTM networks for Automatic Music Composition", 1st Conference on Computer Simulation of Musical Creativity, 2016

[12] Huang, A., and Wu, R, Deep Learning for Music arXiv:1606.04930v1,2016

[13] Vitelli M. and Nayebi A. "Algorithmic Music Generation using Recurrent Neural Networks", 2015. GRUV

[14] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, Koray Kavukcuoglu WaveNet: A Generative Model for Raw Audio arXiv:1609.03499v2, 2016

[15] J.-P. Briot, G. Hadjeres, F. Pachet, Deep learning techniques for music generation – a survey, arXiv preprint arXiv:1709.01620, 2017.

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

[17] R.P. Winnington-Ingram, Ancient Greek music: a survey,

MusicLett. X (1929), 326–345.

[18] G. Nierhaus, Algorithmic Composition: Paradigms of Automated Music Generation, Springer, New York, USA, 2009.

[19] M. Müller, Fundamentals of Music Processing: Audio, Analysis,Algorithms, Applications,唭Springer, 2015.

[20] A. Dejrolo, Acoustic and Midi Orchestration for the Contemporary Composer, Focal Press, Burlington, 2009.

[21] Chun-Chi J. Chen and Risto Miikkulainen, "Creating melodies with evolving recurrent neural networks", 2001 International Joint Conference on Neural Networks, 2001.

[22] Dr.P.SHOBHA RANI1, M.J.SATHISH4, "MUSIC GENERATION USING DEEP LEARNING", International Research Journal of Engineering and Technology (IRJET), Mar 2020

[23] I-Ting Liu and Bhiksha Ramakrishnan. Bach in 2014: Music composition with recurrent neural network. Under review as a workshop contribution at ICLR 2015, 2015