International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 03 | Mar 2020 www.irjet.net p-ISSN: 2395-0072

MUSIC GENERATION USING DEEP LEARNING

Dr.P.SHOBHA RANI[1], S.V.PRANEETH[2], VATTIGUNTA REVANTH KUMAR REDDY[3], M.J.SATHISH[4]

[1]Assistant Professor, Department of Computer Science Engineering, RMD Engineering College, Tamil Nadu, India
[2,3,4]Student, Department of Computer Science Engineering, RMD Engineering College, Tamil Nadu, India

Abstract - *Composing music is a very interesting challenge that tests the composer's creative capacity, whether it a human or a computer. Although there have been many arguments on the matter, almost all of music is some regurgitation or alteration of a sonic idea created before. Thus, with enough data and the correct algorithm, deep learning should be able to make music that would sound human. This report outlines various approaches to music composition through Neural Network model, it is evident that musical ideas can be gleaned from these algorithms in hopes of making a new piece of music. The use of deep learning to solve problems in literary arts has been a recent trend that has gained a lot of attention and automated generation of music has been an active area. This project deals with the generation of music using some form of music notation relying on various LSTM(Long Short Term Memory) architectures. Fully connected and convolutional layers are used along with LSTM's to capture rich features in the frequency domain and increase the quality of music generated. The work is focused on unconstrained music generation and uses no information about musical structure such as notes or chords to aid learning.*

Key Words: Neural Networks, LSTMs, Convolutional layers, fully connected layers, notes, chords.

1. INTRODUCTION

This paper focuses on generating music automatically using Recurrent Neural Network(RNN).We do not necessarily have to be a music expert in order to generate music. Even a non-expert can generate a decent quality music using RNN. We all like to listen interesting music and if there is some way to generate music automatically, particularly decent quality music then it's a big leap in the world of music industry.

Task: Our task here is to take some existing music data then train a model using this existing data. The model has to learn the patterns in music that we humans enjoy. Once it learns this, the model should be able to generate **new** music for us. It **cannot simply copy-paste** from the training data. It has to understand the patterns of music to generate **new** music. We here are not expecting our model to generate new music which is of professional quality, but we want it to generate a **decent quality music** which should be **melodious** and **good to hear**.

1.1 Recurrent Neural Network (RNN)

Now since our music is a sequence of characters therefore the obvious choice will be RNN or variations of RNN like LSTMs or GRUs which can process sequence information very well by understanding the patterns in the input.

A recurrent neural network is a specific type of neural network which is based on sequential information and data. Reason behind calling them recurrent because same set of weights are applied recursively over a differential graph-like structure. RNN has great application in Natural Language Processing.

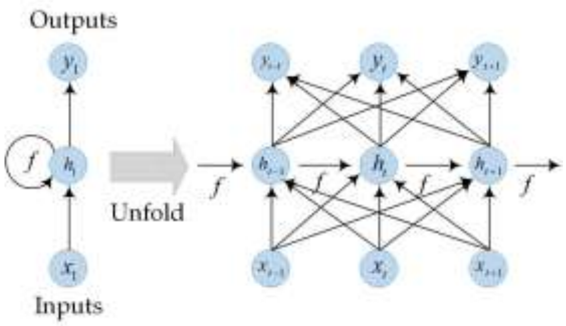

Fig -1: RNN structure

1.2 Char-RNN model

There is a special type of RNN called char RNN. Now our music is a sequence of characters. We will feed one after the other character of the sequence to RNN and the output will be the **next character** in the sequence. So, therefore, the number of output will be equal to the number of inputs. Hence, we will be using **Many-to-Many RNN**, where number of output is equal to the number of input.

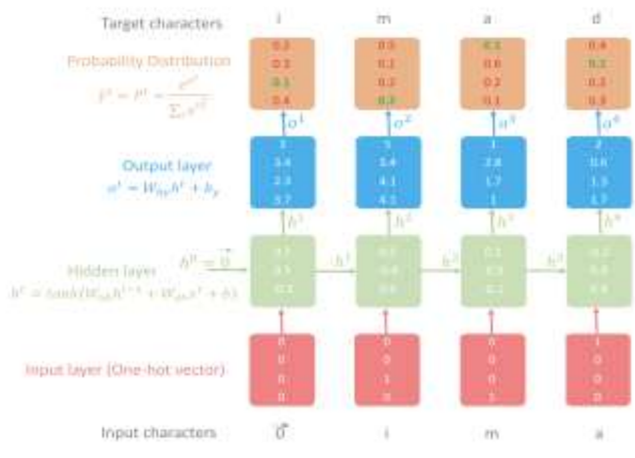

FIG -2: Char-RNN structure

2. METHOD

2.1 Training data (or) Input Data

There are many ways to represent music such as sheet music, ABC notation, MIDI notation, etc. In this paper we would demonstrate the music generation by using ABC notation as our training set as it is easy to understand and easy to represent using just sequence of characters.

```
<score lang="ABC">
X:1
T:The Legacy Jig
M:6/8
L:1/8
R:jig
K:G
GFG BAB | gfg gab | GFG BAB | d2A AFD |
GFG BAB | gfg gab | age edB |1 dBA AFD :|2 dBA ABd |:
efe edB | dBA ABd | efe edB | gdB ABd |
efe edB | d2d def | gfe edB |1 dBA ABd :|2 dBA AFD |]
</score>
```

Part-1

Part-2

Fig -3: Sample ABC notation of music

2.2 Data Preparation

We will feed data into batches. We will feed batch of sequences at once into our RNN model. First we have to construct our batches.

We have set following parameters:

Batch Size = 16

Sequence Length = 64

We have found out that there are total of **155222** characters in our data. Total number of unique characters are **87**.

We have assigned a numerical index to each unique character. We have created a dictionary where key belongs to a character and its value is it's index. We have also created an opposite of it, where key belongs to index and its value is it's character.

2.3 Batch Construction

There are a total of 155222 characters in our data set, out of which only 87 characters are unique. We have assigned index to each of the characters. So the below numbers in the batch are not the exact numbers. In reality the batches will contain index of the corresponding character.

	Batch-1	Batch-2	...	Batch-150	Batch-151
0	0...63	64...127	...	9536...9599	9600...9663
1	9701...9764	9765...9829	...	19237...19300	19301...19364
.	...	...	...	...	...
.	...	...	...	...	...
.	...	...	...	...	...
14	135814...135877	135878...135941	...	145350...145413	145414...145477
15	145515...145578	145579...145642	...	155051...155114	155115...155178

2. 4 Network Architecture

Our model consists of these layers.

● LSTM: A Recurrent Neural Network

A traditional human brain learns by persisting things. We do not start learning by scratch every time instead we learning things above what we have already learned. A traditional neural network cannot do this. For example, to understand the events happening at every point in a movie. Recurrent Network are networks with loop in them.

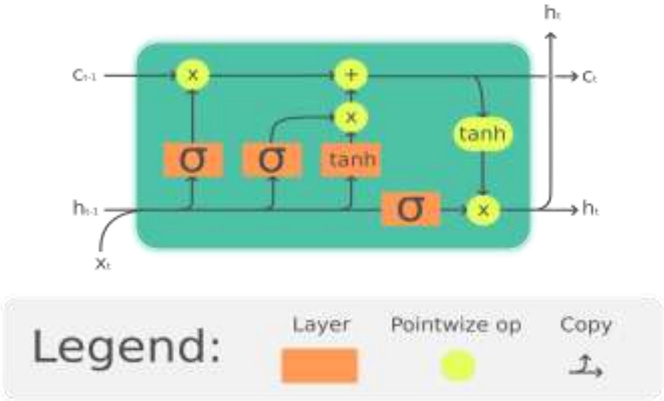

FIG-4: LSTM cell

● Dropout layers

To reduce overfitting in neural network we are using Dropout technique.

● Dense layers is nothing but a fully connected neural layer connecting each input node to output node.

● The Activation layer

Activation functions determine the output of a deep learning model, its accuracy, and also the computational efficiency of training a model.

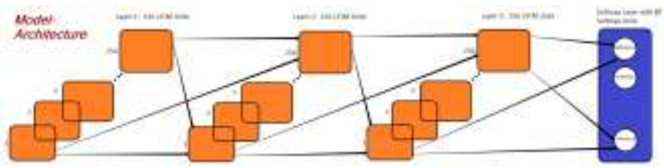

FIG -5: Network Architecture

2.4 Music Generation

Now, we have already trained our model and find the best weights. Now our model is ready for making prediction. In order to make prediction, we will give any of the 87 unique characters as input to our model and finally it will generate 87 probability values through softmax layer. From these returned 87 probability values, we will choose the next character probabilistically and not deterministically and finally we will again feed the chosen character back to the model and so on. We will keep on concatenating the output character and generate music of some length. This is how music will be generated.

3. FURTHER SCOPE

We have generated a good quality music, but there is a huge scope of improvement in it.

First, starting and ending music can be added in every new generated tune to give a tune a better start and better ending. By doing this, our generated music will become melodious.

Second, the model can be trained with more tunes. Here, we have trained our model with only 405 musical tunes. By training the model with more musical tunes, our model will not only expose to more variety of music but the number of classes will also increase. By this more melodious and at the same time more variety of music can be generated through the model.

Third, model can also be trained with multi-instrument tunes. As of now, the music generated is of only one piece of instrument. It would be interesting to listen what music the model will produce if it is trained on multi-instrument music.

Finally, a method can be added into the model which can handle unknown notes in the music. By filtering unknown notes and replacing them with known notes, model can generate more robust quality music.

4. CONCLUSION

We used simple LSTM based network to automate music generation. Results may not be perfect but are very good which shows that the neural network can be used to create music and has potential to produce higher complex musical extracts. LSTM proved to be a good model for capturing long-timescale dependencies. By providing musical note objects to our network, it was able to learn a musical style which was then used to generate the music. Future work can be done on the variants of LSTM and ensemble models which will require high powered GPUs.

REFERENCES

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

[2] Hiller, L., Isaacson, L.M.: Experimental Music. Composition with an Electronic Computer. McGraw-Hill Book Company (1959)

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

[4] https://deeplearning4j.org/lstm.htm

[5] https://keras.io/

[6] https://colah.github.io/posts/2015-08-Understanding-LSTMs/