Generative Adversarial Network (GAN) for Irish Traditional Music Generation

Tapan Vivek Auti (20231499)
School of Computer Science
National University of Ireland, Galway

Supervisors
Prof. Mathieu d'Aquin

In partial fulfillment of the requirements for the degree of
MSc in Computer Science (Artificial Intelligence)
August 31, 2021

DECLARATION I, TAPAN VIVEK AUTI, do hereby declare that this thesis entitled Generative Adversarial Network (GAN) for Irish Traditional Music Generation is a bonafide record of research work done by me for the award of MSc in Computer Science (Artificial Intelligence) from National University of Ireland, Galway. It has not been previously submitted, in part or whole, to any university or institution for any degree, diploma, or other qualification.

Signature: ________________________________

Abstract

The earliest research can be found around the 1940's and the majority of the Research that emerged over Neural Network and it's application can be evidently found after the 1980's. From then to present there have been significant changes in the applications and use of Neural Network and the concepts of Deep Learning. Many researches emerged in all fields about machines showing cognitive abilities, which is not new. The capabilities of performing human tasks and cognitive thinking shown by machines is what debatably AI can achieve and researches in. Over the years the research gave rise to new concepts of reproducing various activities performed by human, one such being Music.

The concept of music reproduction with the help of deep learning is as well not new, but what is new is the quality of regeneration, it's similarities to original work and also the uniqueness of melodies generated. We can say the machine truly learnt and performed, along with all these things one crucial thing that is also new in today's research is the availability of resources. This thesis has tried to achieve one such similar thing where I have tried to reproduce Irish traditional folk music.

Generative Adversarial Network model has been proposed for generation where Convolutional Neural Network will be used as generator and as discriminator (...) is used. ABC files have been used to train the model for music generation. The aim of this thesis is to generate tunes that will be similar to Irish Folk Music with a model that is not specifically tuned for Irish Folk but can later be used on any type of music and generate similar tunes.

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

The vast applications of Deep Learning in fields of computer vision, analysis, prediction, etc. are no doubt noteworthy but, the implementations and modelling used for generation like the one in WOMBO app for music lip syncing or Instagram filters have always intrigued me. The power of AI to be used to learn and reproduce something completely unique yet similar is one of the greatest research goals that we have achieved over the years.

Previously, having seen many applications of Deep Learning being used to reproduce melodies of various individual instruments or random noise in form of music made me realise that the scope for this and other variants are unprecedented, also having a bit of predefined interest for Folk Music I decided to use Traditional Irish Music for my thesis and to reproduce melodies that showed resemblance to the same.

The outcome I wish to achieve, are melodies generated that are not exactly sounding like one of our input songs used or like one copied from the composer directly with some minor changes, but more or less matching the same genre or theme, with the tunes being pleasant to hear and sounding like Irish Traditional Music to people hearing it for the first time. The data and model to be built on should be something which was not exactly done before with the properties of Robustness, Flexibility and Polymorphic in nature.

1.2 Data Used

For the purpose of this thesis I researched various repositories freely available on the internet that had a clear focus on Irish Traditional Music. After thorough research we decided to go with the repository made freely available by Jeremy (2017).

The music files available on this site are in ABC format. Normally ABC format is converted into midi format that is then converted into .wav which we can hear as normal music. A basic ABC file looks like the following.

```
X:1
T:Paddy O'Rafferty
C:Trad.
M:6/8
K:D
dff cee|def gfe|dff cee|dfe dBA|dff cee|def gfe|faf gfe|1 dfe dB
A:|2 dfe dcB|]
~A3 B3|gfe fdB|AFA B2c|dfe dcB|~A3 ~B3|efe efg|faf gfe|1 dfe dcB:
2 dfe dBA|]
fAA eAA|def gfe|fAA eAA|dfe dBA|fAA eAA|def gfe|faf gfe|dfe dBA:|
```

-Walshaw (2010)

The ABC file has two parts, the header that has all the basic information including various other parameters depending on the type of file and the second part are the notes, In the example above the first five lines are the header and the last 3 lines are the notes.

```
X:1, is the reference number,
T:Paddy O'Rafferty, is the Title,
C:Trad, defines the composer,
```

here it means traditional as the original composer name is lost due to the tune being vintage so its better to mark this as traditional, this varies according to the creator of the file. A lot of other features are present but the ones that concern us are only X,M,K and L where M and K are the meter and key of tune, X contains the reference number that is not that important and L is note length, L being the most important as this will help us split the tune into parts if we want in preprocessing of the data. More detailed information related to these notations and type of file is given in Chapter 2 - Background.

1.3 Flow of Thesis

Chapter 1 - Introduction, gives a brief introduction to the thesis, a generic idea of the data being used, the motivation for the thesis and flow of thesis.

Chapter 2 - Background, gives thorough information about all the technologies that are being used in this thesis and tries to explain every concept used in brief, along with detailed explanation of the data being used and any other relevant information related to the dataset.

Chapter 3 - Related Work, In this chapter I have tried to cover all the prominent research from the past that match my work and tried to explain briefly about how the advancements were made over the years till now and what their approaches were.

Chapter 4 - Methodology, In this chapter I have tried to give a step by step and detailed information about all the steps conducted to complete the modelling of thesis and its implementation, right from the data preprocessing to the models used and the training done to achieve results.

Chapter 2

Background

2.1 Deep Learning

IBM (2020) defines deep learning as ”Deep learning attempts to mimic the human brain—albeit far from matching its ability—enabling systems to cluster data and make predictions with incredible accuracy.”

Deep Learning is a subfield of machine learning that deals with model or algorithms that are such constructed that they mimic the structure of our brain, basically a type of Neural Network with a lot of layers of hierarchy but always not necessarily many, with each level having some sense of abstraction that deals with the huge amount of data and processes it. There are Neural Networks with just one layer too but the addition of layers helps deal with the complexity and to get better results.

The applications of Deep Learning are way beyond imagination and the future scope where we can implement it is without question substantial.

The following is the most basic and simple representation of a Neural Network.

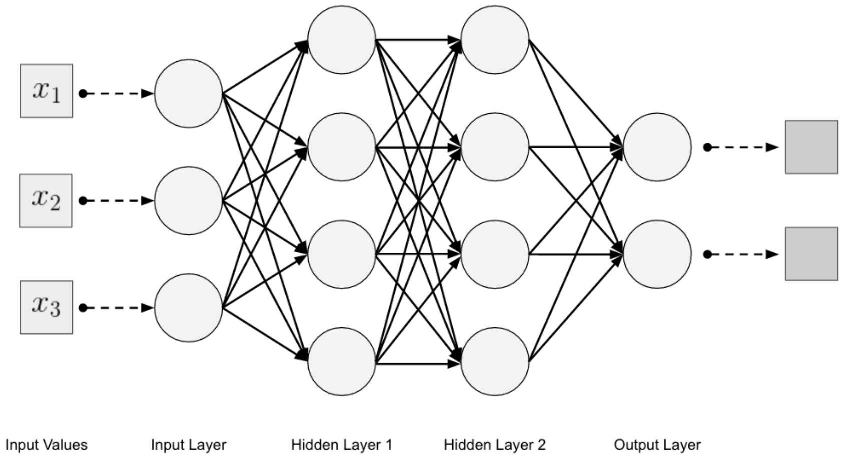

Figure 2.1: Deep Neural Network

Deep Learning uses various factors such as inputs , weights and biases that are processed upon to get accurate predictions and classifications, etc. The Deep Neural Network have multiple layers and each layer has multiple nodes, these layers are built upon each other to get the most optimised solution.

2.2 Recurrent Neural Network

IBM (2020), ”A recurrent neural network (RNN) is a type of artificial neural network which uses sequential data or time series data.” Unlike Feedforward Neural Network, RNN have their own internal memory, it uses the same function for each input and the output depends on the previous operations, the output produced is then sent back to RNN. It learns from its previous inputs and makes prediction upon them.

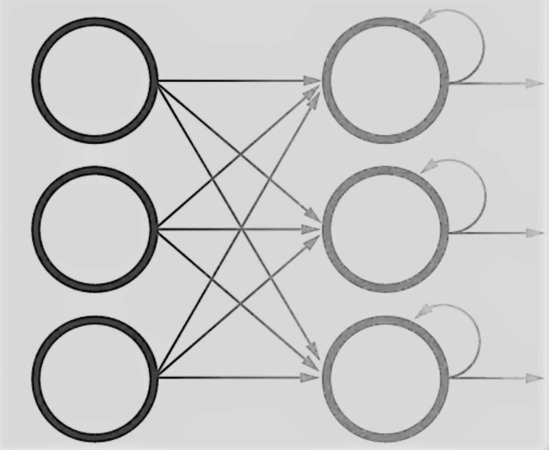

Figure 2.2: Recurrent Neural Network

So basically the next step has two inputs, the output of the current state as well as the input of the state and so on. An activation function determines whether the neuron should be activated and it brings the activations to an range depending upon the functions used. The most common functions are-

- Sigmoid Function:

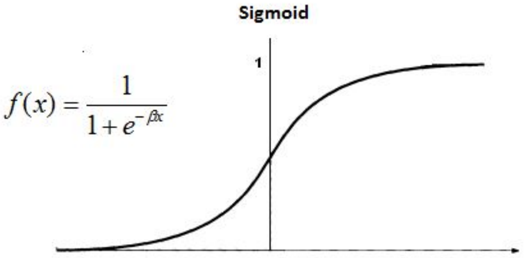

Figure 2.3: Sigmoid Function

- Relu Function:

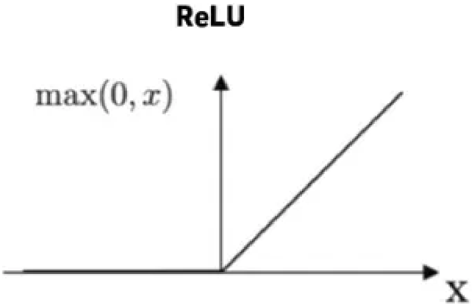

Figure 2.4: Relu Function

- Tanh Function:

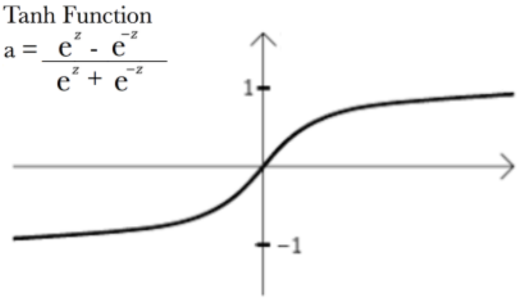

Figure 2.5: Tanh Function

2.3 LSTM (Long-Term Short Memory)

LSTM is a popular Recurrent Neural Network(RNN) Architecture, it solves the problem of long-term dependencies of RNN. In cases where the current state's output is being affected by the input of states occurring much earlier in the flow, then in such case it becomes very hard for the RNN to correctly predict the output and RNN fails. To overcome this LSTM was introduced, it was first used to solve vanishing gradients problems and unlike RNN it has three states, input, output and a forget gate which controls the flow and give predictions.

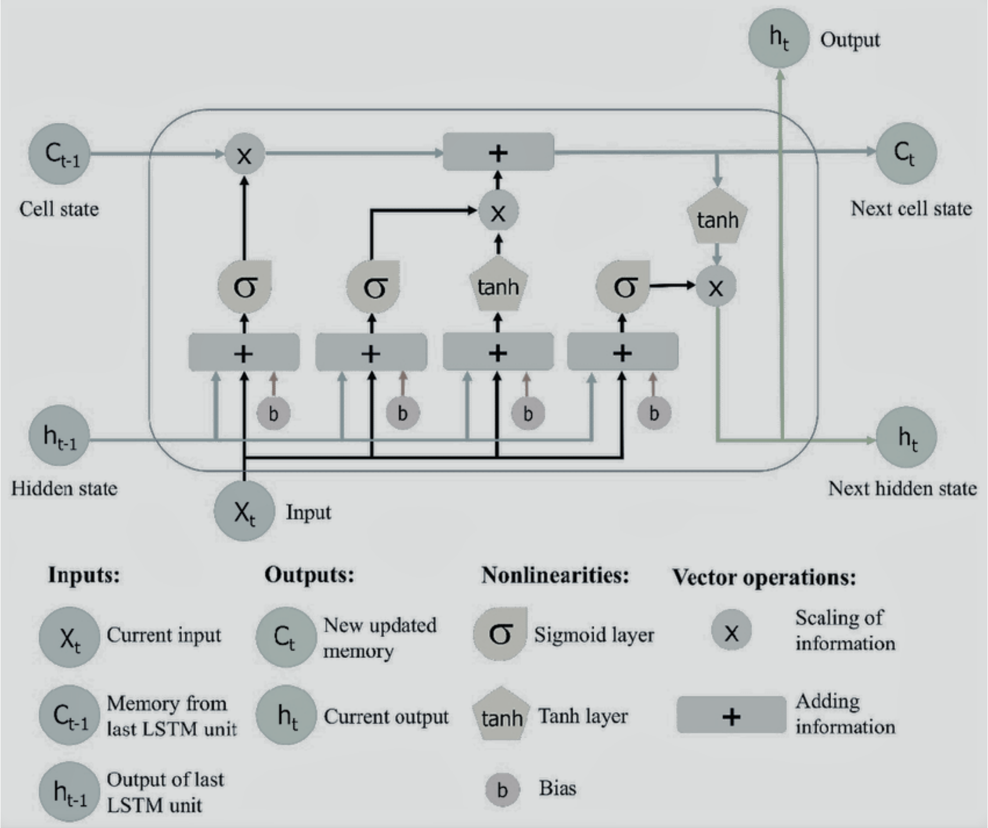

Figure 2.6: LSTM Structure by YAN (2016)

2.4 CNN (Convolutional Neural Network)

Three main reasons that make CNN better than any other Neural Network is the convolutional layer, the pooling layer and the fully connected layer. With each layer the CNN's complexity is increased, the first few layers focus on simple features and the later on more complex identification etc. CNN have much greater performance than other with respect to image, text and audio files. The main building block of ConvNets is the convolutional layer, all the main computations occur here. It has 3 main parts i.e. filter, feature map and the input data. IBM (2020)

Pooling is basically down sampling the input and deals with the dimensionality reduction of the parameters from the input. IBM (2020) There are two main types of pooling, max pooling and average pooling.

The last Layer is the fully connected, this layer all the nodes in output layer are connected to the previous layer, this is normally the last layer of the model.

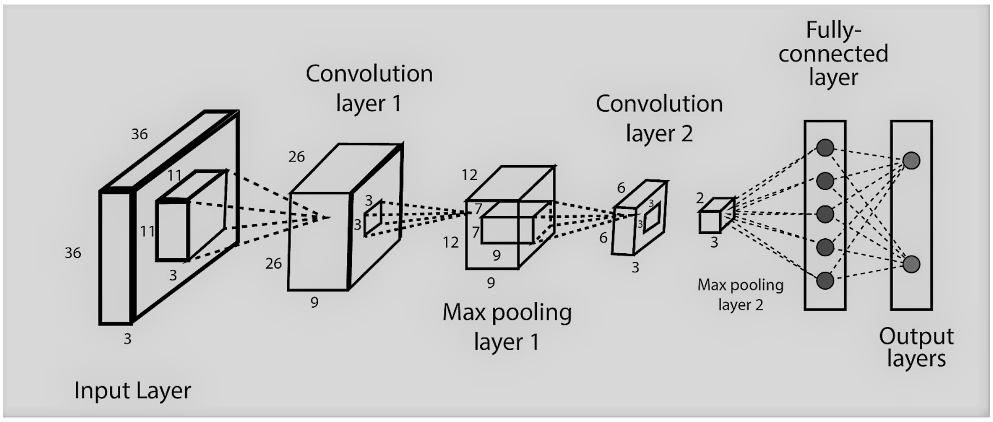

Figure 2.7: ConvNets Architecture by Balodi (2019)

2.5 GAN (Generative Adversarial Netwrok)

This is a similar concept to that of Convolutional Neural Network, it is a type of generative modelling. Generative modelling is a subfield of machine learning where the features are found and learnt upon automatically from the input data in a way that new output can be generated from the model.

There are two parts of GAN, one being the generator that creates new examples from the data and the discriminator that classifies the data, the job of the generator is to create such examples that the discriminator is not able to identify correctly and discriminator tries to find the real ones and the fake ones. The two models are run simultaneously.

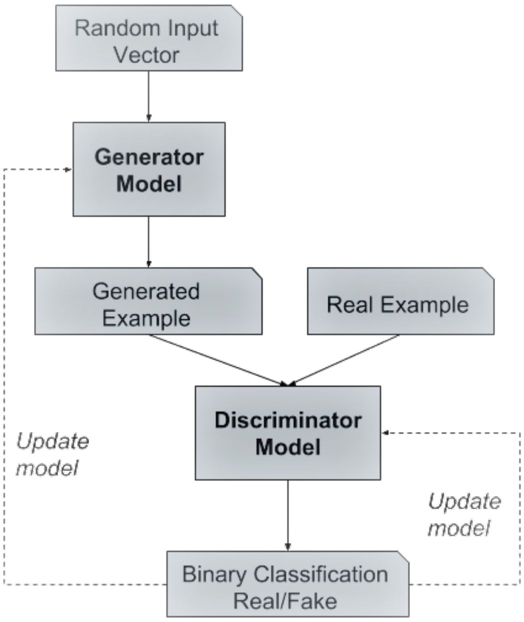

Figure 2.8: Architecture of GAN as defined by Brownlee (2019)

2.6 Keras

Chollet (2015) defined ”Keras is an API designed for human beings, not machines. Keras follows best practices for reducing cognitive load: it offers consistent and simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear and actionable error messages. It also has extensive documentation and developer guides.”

Basically Keras is a deep learning framework focusing on Neural Networks, it uses a python interface for the same. It is mostly used to deal with the various aspects of building models of Neural Network like the layers and functions, it has support to Convolutional and Recurrent Neural Network as well.

2.7 Tensorflow

Tensorflow is more of an open-source library with which we can perform various Machine Learning tasks.

Brain (2015) defined it as ”TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in Machine Learning and developers easily build and deploy Machine Learning powered applications.”

It gives us the ability to build and train Machine Learning models easily, and also deploy them on cloud. It was developed by Google Brain team for internal use and used for research purposes in Google. We can debug and train keras and tensorflow code using tensorflow, it is mostly used for numerical computations using graphs.

2.8 Data Representation

The ABC notations use a-g and A-G, Z for representation of notes and rests, there are other elements that denote the type of notes i.e. sharp, flat, raised or lower octave, the length and ornamentation.

High notes are depicted with " ' ", and low notes are depicted with " , " . The long notes are shown with dash or hyphen " - ". The bar line "|" is used to separate the bars i.e. the beat of the music.

Chapter 3

Related Work

The earliest work which used Neural Network for music composition, analysis and generation can be found around the late 1980's, Dolson (1989). This paper dealt with the traditional ways that were used for music generation, and how Neural Network can be used to help achieve greater results. Here the author used a simple implementation for music generation which used a real-time recurrent learning algorithm to generate four simple rhythms and also explained in depth about back-propagation and generalization in terms of music generation.

What has changed from then to today, is the availability of more discrete resources for data gathering and the variations that can be found in the availability and type of data. Before, Musical data was available in non symbolic forms i.e. musical notes (polyphonic music) ,etc. It was very hard to train and was complex in training aspects due to scarce availability of data. "The availability of resources has paved a new pathway for more advancements in deep learning", Deng and Yu (2014), and hence a lot of research has been conducted since then regarding music generation with Neural Network.

Pons (2015) suggests that all earlier work was done using a single melody, that means only a single instrument was used or only one note was used to generate music. Piano rolls were amongst the most common one's used. The majority of the research being conducted for music generation was done using RNN and LSTM, Boulanger-Lewandowski et al. (2014), Chen and Miikkulainen (2001), Eck and Lapamle (2008). Chen and Miikkulainen (2001) did evaluation based on the contraints of the output melodies. They combined different aspects such as pitch, notes and based upon their respective weight defined whether the structure of the melody matched the genre of the song.

The very first one's to use RNN was MOZER (1994). He provided motivation for Eck and Schmidhuber, who later were the first ones to apply LSTM for music generation, Eck and Schmidhuber (2002). They thought that the music generated from RNN lacked a definite global structure and that it could not learn an entire musical form, the reason behind that being, the temporary events were not handled well by RNN and hence they thought, to deal with these issues there should be something more discrete, so they used LSTM, as RNN also had other drawbacks like failing at timing, counting and CSL learning which were removed using LSTM. Eck and Lapamle (2008) did the evaluation of the model and generated music by plotting the probability of selection between chords and notes. According to him if the plot shows fixed length repeated loops then it is aimless music and the one which incorporates several structures is good music.

Sturm et al. (2016) did the most significant use of sequential ABC files for music generation using LSTM, they in the paper suggest two models for it ,one is Char RNN which uses a single character vocabulary and the other is Folk RNN which used transcription token for generation. They were the first one's to successfully use ABC files for music generation. They preprocessed and separated tunes based on single and transcription tokens and then passed them to the beforementioned models to obtain two different results. Till date the only type of research done on music regeneration was mostly on piano rolls which uses midi files for regeneration , Yang et al. (2017), Dong et al. (2017). These piano rolls generally comprise of tunes from a single instrument and rarely have tunes from multiple instruments. This made a stagnancy in type of music generated, as more or less the dataset used as input was similar. The models used differ in all papers which to give rise to variations in the music produced but still somehow show a sense of similarity when mapped with the melody's. Very few models and papers were such that used a combination of multiple instruments and completely new models which produced unique and never before heard melodies. Sturm et al.

(2016) did the evaluation of the music generated by two methods, one by doing a statistical analysis which compared the statistical output of the tunes with their training data. In this they included the modes, meters, pitch and token length as parameters. The second way was music analysis in which they uploaded their music online and let people decide how it sounded and whether it matched the Genre of the input data used.

Dong and Yang (2018) and various other models were generally polyphonic music generation models. These generated polyphonic music by taking midi files as input. Till late 2018 Mogren (2016) was the only model that used GAN for music generation. He used midi files as input, but had drawbacks of not being able to generate music using priming melody or chord sequences. These were removed in Yang et al. (2017). MIDINET used CNN as their main base model which gave them better results Yang et al. (2017). Yang et al. (2017) purely used only human resources to analise the melodies generated. They made people hear the music generated by the model and the original input used and asked people if they could differentiate between the genre ofthe melodies. All the test subjects used were from musical background.

After various researches it was proved that CNN is faster than RNN and also easily parallelizable van den Oord et al. (2016a), before this maximum models used RNN only. All the years of research has paved path for various genres of music generations, like Tokui (2020) that tried to reproduce rhythm patterns of electronic music using GAN. It takes midi files of EDM and tried to reproduce this music which sounded like one that is produced on a professional instrument and resembled actual EDM to some extent. Tokui (2020) used rhythm similarities to distinguish and evaluate their model. They plotted the matrix produced by comparing the Rhythm Patterns.

Like symbolic data for generation, various studies were also conducted for music generation through non symbolic forms such as raw music files and wave forms. van den Broek (2021) was one such successful model that generated music from raw files. They used low convolutional GAN as their base model and achieved a milestone of having extremely low computational power while generation. van den Broek (2021) used three factors for evaluation, the first one was the audio quality and timbre defined by the tonality of the tune, the second was the musicality, this is the rhythmic structure of the tune and the last one was sample diversity based on the tempo.

All the models that used GAN had one major drawback, that due to no or less temporal feature extraction the generated music didn't always sound natural and was unstable to overcome. Guan et al. (2019) proposed a new model that considered temporal features as well and had a self-attention mechanism to enable GAN and introduced a new method of switchable normalization to stabilize network training giving them very good results and music generated being stable and soothing to ear.

A lot of research is being made in music generation considering various aspects of music such as music generation based upon the genre or feel of music, one such proposed model is used in Huang and Huang (2020). Here the authors took a dataset that has emotions tagged along with the song in the input files, so while producing output the model trains on basis of the feel of the music and generates music based on what genre you want like rowdy, romance, quiet, majestic etc. Huang and Huang (2020) also used human subjects for identifying the resemblence between the tune and emotions.

Engel et al. (2019) was the most recent work that successfully synthesised audio by using wave files as input and using GAN for processing and generation. They used a specific controlled dataset for music generation and were able to achieve great speed which was claimed to be one thousand times faster generation than that of van den Oord et al. (2016b).

The latest research going on has been on music generation from lyrics, Yu et al. (2021) used LSTM generator and discriminator for generation, where in the dataset the lyrics are mapped to melodies, but comparatively such data is still very scarce to find and hence the result of the mapping is not melodies to lyrics, i.e. the generated music and lyrics not always correlate and sound good to hear, its very random and doesn't make any sense and generally sounds like music with a lot of noise in it and totally irrelevant to lyrics.

Chapter 4

Methodology

4.1 Data Preprocessing

- Convert the ABC file to string and join all songs into a single string file.
- Find all unique characters.
- Vectorize the text by creating two lookup tables -
  - The first lookup will map the letter to numbers
  - The second lookup will map the number to individual letters
- Convert the String of unique character from the list of all characters to a Vectorized form.
- Use mapping to convert the vocabulary characters (string) to corresponding indices.
- The output should be numpy array with N elements where N is the length of input string.

4.2 Implementation

Yet to finalize

4.3 Evaluation and Results

- The statistical analysis of the factors of the melodies will be compared to those from the original data. This will include timbre, pitch, mode, rhythm and temporal features of the tunes.

- Human subjects will be used for analysis, a questionaire will be used to take people's opinion on the tune generated and its resemble to Irish Folk.

References

Tanesh Balodi. Convolutional neural network (cnn): Graphical visualization with python code explanation, Sep 2019. URL https://www.analyticssteps.com/blogs/convolutional-neural-network-cnn-graphical-visualization-code-explanation. iv, 8

Nicolas Boulanger-Lewandowski, Gautham J. Mysore, and Matthew Hoffman. Exploiting long-term temporal dependencies in nmf using recurrent neural networks with application to source separation. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6969–6973, 2014. doi: 10.1109/ICASSP.2014.6854951. 12

Google Brain. Tensorflow web guide. https://www.tensorflow.org/, 2015. 10

Jason Brownlee. What are gan, Jun 2019. URL https://machinelearningmastery.com/what-are-generative-adversarial-networks-gans/. iv, 9

Chun-Chi J. Chen and Risto Miikkulainen. Creating melodies with evolving recurrent neural networks. In *Proceedings of the INNS-IEEE International Joint Conference on Neural Networks*, pages 2241–2246, Piscataway, NJ, 2001. IEEE. URL http://nn.cs.utexas.edu/?chen:ijcnn01. 12

François Chollet. Keras web guide. https://keras.io/, 2015. 10

Li Deng and Dong Yu. Deep learning: Methods and applications. *Foundations and Trends in Signal Processing*, 7(3–4):197–387, 2014. ISSN 1932-8346. doi: 10.1561/2000000039. URL http://dx.doi.org/10.1561/2000000039. 12

M. Dolson. Machine tongues xii : Neural networks. *Computer Music Journal*, 13:28–40, 1989. 12

Hao-Wen Dong and Yi-Hsuan Yang. Convolutional generative adversarial networks with binary neurons for polyphonic music generation, 2018. 14

Hao-Wen Dong, Wen-Yi Hsiao, Li-Chia Yang, and Yi-Hsuan Yang. Musegan: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment, 2017. 13

Douglas Eck and Jasmin Lapamle. Learning musical structure directly from sequences of music, 2008. 12, 13

Douglas Eck and Jürgen Schmidhuber. Learning the long-term structure of the blues. In José R. Dorronsoro, editor, *Artificial Neural Networks — ICANN 2002*, pages 284–289, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg. ISBN 978-3-540-46084-8. 13

Jesse Engel, Kumar Krishna Agrawal, Shuo Chen, Ishaan Gulrajani, Chris Donahue, and Adam Roberts. Gansynth: Adversarial neural audio synthesis, 2019. 15

Faqian Guan, Chunyan Yu, and Suqiong Yang. A gan model with self-attention mechanism to generate multi-instruments symbolic music. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–6, 2019. doi: 10.1109/IJCNN.2019.8852291. 15

Chih-Fang Huang and Cheng-Yuan Huang. Emotion-based ai music generation system with cvae-gan. In *2020 IEEE Eurasia Conference on IOT, Communication and Engineering (ECICE)*, pages 220–222, 2020. doi: 10.1109/ECICE50847.2020.9301934. 15

IBM. Ibm deep learning. https://www.ibm.com/cloud/learn/deep-learning, 2020. 4, 5, 8

Jeremy. The session. https://thesession.org/tunes, 2017. 2

Olof Mogren. C-rnn-gan: Continuous recurrent neural networks with adversarial training, 2016. 14

MICHAEL C. MOZER. Neural network music composition by prediction: Exploring the benefits of psychoacoustic constraints and multi-scale processing. *Connection Science*, 6(2-3):247–280, 1994. doi: 10.1080/09540099408915726. URL https://doi.org/10.1080/09540099408915726. 12

Jordi Pons. Deep learning for music information research. Technical report, Universitat Pompeu Fabra, 2015. 12

Bob Sturm, João Santos, Oded Ben-Tal, and Iryna Korshunova. Music transcription modelling and composition using deep learning. *arXiv*, 04 2016. 13

Nao Tokui. Can gan originate new electronic dance music genres? – generating novel rhythm patterns using gan with genre ambiguity loss, 2020. 14

Korneel van den Broek. Mp3net: coherent, minute-long music generation from raw audio with a simple convolutional gan, 2021. 15

Aaron van den Oord, Nal Kalchbrenner, Oriol Vinyals, Lasse Espeholt, Alex Graves, and Koray Kavukcuoglu. Conditional image generation with pixelcnn decoders, 2016a. 14

Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alexander Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio. In *Arxiv*, 2016b. URL https://arxiv.org/abs/1609.03499. 16

Chris Walshaw. how-to-understand-abc-the-basics, 2010. URL https://abcnotation.com/blog/2010/01/31/how-to-understand-abc-the-basics/. 2

YAN. Understanding lstm's and its diagram, 2016. URL https://blog.mlreview.com/understanding-lstm-and-its-diagrams-37e2f46f1714. iv, 7

Li-Chia Yang, Szu-Yu Chou, and Yi-Hsuan Yang. Midinet: A convolutional generative adversarial network for symbolic-domain music generation, 2017. 13, 14

Yi Yu, Abhishek Srivastava, and Simon Canales. Conditional lstm-gan for melody generation from lyrics. *arXiv*, 17(1), 2021. ISSN 1551-6857. doi: 10.1145/3424116. URL https://doi.org/10.1145/3424116. 16