Algorithmic Music Composition Using Probabilistic Graphical Models and Artificial Neural Networks

Marc Marsden
School of Computer Science and Applied Mathematics
The University of the Witwatersrand
Johannesburg, South Africa
Email: 1437889@students.wits.ac.za

Ritesh Ajoodha
School of Computer Science and Applied Mathematics
The University of the Witwatersrand
Johannesburg, South Africa
Email: ritesh.ajoodha@wits.ac.za

***Abstract*—Composing music algorithmically has been a goal long-pursued by many computer scientists. Various methods have been implemented to achieve this, ranging from randomly selecting musical components to deep learning models. The main focus of this research is to develop a model which fools a human into believing the output music is human-made. This study uses for this a collection of rock music MIDI files which the notes, chords, pitches and duration are extracted as features. A Bayesian network is selected as the main model for this research and a Long Short-Term Memory (LSTM) network as the benchmark model. A Turing test was performed on 20 people for both models and the LSTM on average was identified as human-made 36% of the time, while the Bayesian network, on average, had been misidentified 39% of the time. These results may indicate that music is more probabilistic than time-dependent.**

***Index Terms*—Bayesian networks, LSTMs, Music, Composition, Machine Learning.**

I. Introduction

Algorithmic music composition refers to the utilisation of formal procedures to create music with minimal human involvement. Before the computer, 18th century musicians developed a game known as "musical dice game" which involved piecing together randomly selected fragments of music; creating new compositions based only on chance [1]. This interest in making music without human intervention has only grown more over time. Today, various disciplines such as the music industry, telecommunications, artificial intelligence, and philosophy research music generated by algorithms so they may better understand and improve their respective fields [2].

Due to its long history and application in many different fields, countless literature exists, especially in the field of machine learning. Many machine learning models have been successful at generating music. The purpose of this research is to provide a data driven tool which implements Bayesian techniques to compose music.

There have been many attempts to implement a model which is indistinguishable from human-made music [3], [4], [5]. In [3], the complex temporal structure of music is discussed and how previously developed deep learning models failed to capture this essential factor. To resolve this, the paper adopted a more complex deep learning model which had interesting results.

Similar models have been used and compared to a rule-based algorithm [5]. Two experiments were performed on each algorithm, using a different genre for each experiment. The results indicated a better performance in the rule-based model than the deep learning model. Instead of a deep learning model, researchers developed a probabilistic graphical model with a genetic algorithm [4]. This was in an attempt to provide a new method of algorithmically generating music.

The research design involves three main sections: preprocessing, model training, and evaluation. Preprocessing involved the extraction of the chosen features from the data in addition to formatting the data for the input of the models. The training of the models was where the extracted data had been fitted to the models. The final phase involved the model generating its output along with a survey which was conducted to determine the quality of the output.

The following contributions are made: a) We provide a music generating Bayesian network model which fools a human into thinking its output is human-made. b) We provide a music generating Bayesian network model which outperforms a Long Short-Term Memory (LSTM) network in a Turing test.

Section II presents a review of the background and related work needed for this research. Section III explains the methodology and the algorithms used to compose and evaluate the music. Results and discussion are in Section IV and V, respectively. Finally, Section VI contains the conclusion and suggestions for future usage.

II. Related Work

Since the days of Pythagoras, the relationship between music and mathematics has been documented. Various disciplines such as the music industry, telecommunications, artificial intelligence and philosophy, research music generated by algorithms so they may better understand and improve their respective fields [2].

In this section, we will present the works of previous researchers and how they have attempted to algorithmically compose music as well as how they evaluated said music.

A. Data

In the many papers written on the subject of algorithmic music composition, the two main files which were used to

train the models were Musical Instrument Digital Interface (MIDI) files and raw audio files (WAVE). Standard MIDI files contain an almost symbolic representation of music and most are structured as several tracks [6]. A WAVE file is a raw audio format created by Microsoft and IBM. The WAVE format is uncompressed lossless audio which stores audio data, time signals, track numbers, sample rates as well as bit rate [7]. The core difference between these two file formats is that a WAVE file is many times larger than a MIDI file [8].

The MIDI file contains two very important tracks for music composition, the melody track and the accompaniment track. This allows for a more music theory approach. In [5], two methods of algorithmic composition using rock and jazz music are presented and compared. Midi events, namely message types, notes, velocity, and time - were extracted as features, and all the music notes contained in the MIDI files were collected and separated into note objects and chord objects. By combining the features from the midi events with the note and chord objects, the octave interval can be calculated for each note, which will allow for the creation of a melody track and accompaniment track. This shows that, if one wants to create an algorithm that adheres to musical theory guidelines, a MIDI file is the best choice, since it is fundamentally a digital music sheet.

Many people lack the theoretical knowledge of music to take full advantage of the information stored in a MIDI file. However, one may create a model to generate music without information on musical structure or theory by utilizing raw audio files in the frequency domain [9]. To be able to work in the frequency domain, the continuous-time signals are converted into discrete-time signals. The signals are then divided into small blocks–the size of the sampling rate–and converted into the frequency domain using the Fourier transform. Working with music in the frequency domain is an interesting choice since musical notation used to represent music is inherently based on separating sounds into their frequency components or more commonly known as their musical notes.

B. Models

Artificial Neural Networks (ANNs) are a group of models, which have been shown, in many studies, to be highly effective in solving problems of function approximation, classification, clustering, and pattern recognition [10]. When used as generative models, ANNs have produced artistic and non-artistic data almost identical to the real data or human art [11].

Previous studies attempted to extract and generate complex temporal structures of music with the implementation of neural networks, yet were not sufficient enough to truly capture the long timescale musical structure [11]. Recurrent Neural Networks (RNNs) were a possible solution to this problem, since RNNs exhibit temporal dynamic behaviours. However, when applied to music composition, the model performs poorly [3]; possibly due to the vanishing gradient problem. To overcome this shortcoming, the LSTM was developed [12].

LSTMs have been demonstrated to be able to learn both the global structure and the local structure from a collection of musical text training data (using the blues genre) and generate music of the same form [3].

In [5], the work improves the use of LSTMs by implementing the network to generate rock and jazz music with the use of MIDI files instead of sheet music. Fifteen people were asked to identify the genre of several songs generated by the algorithm and, on average, rock was correctly identified 44% of the time and jazz 62.67%.

Bayesian networks are probabilistic models which have been applied to many areas such as text mining, speech recognition, and signal processing [13].

A real-time musical accompaniment system was designed, to receive an acoustic sound and then output a musical accompaniment [14]. A hidden Markov model was used to perform real-time analysis on the received sound and a Bayesian network, consisting of "hundreds" of Gaussian random variables, was developed to produce the accompaniment. In [15], three Bayesian network models were presented. Each of which achieved different music generation tasks (chord voicing, four-part harmonization, and real-time chord prediction).

C. Evaluative Method

The Turing test was developed to assess if a machine was intelligent. A human evaluator is tasked to communicate with a human and a machine, from text only, and then determine which is human. If the machine could trick the evaluator into thinking it was human, the machine was deemed intelligent [16]. Many researchers have modified this to measure the effectiveness of music generating systems. This new framework considers the system successful if the machine output sounds like or is preferred to the human output [17].

The modified test is performed by providing several participants with questionnaires with questions that could range from choosing which song is composed by a human composer [18] to identifying which genre music is playing [5]. A key factor of the test seems to be whether or not the participants know they are comparing human-made music to machine-made music.

In [19], two groups were asked to identify the most "human-like" song out of five. The first group were told one of the pieces was performed by a human, while the second group were told all the songs were algorithmically generated. It was discovered that the first group were less confident in their answers than the second group and the difference in confidence was "statistically significant".

III. Research Design

(i) Rock music MIDI files, each of 30 second duration, collected from various sources online. Each MIDI file is converted to the key of C major/A minor before being preprocessed.

(ii) The notes and chords stored in the MIDI files are extracted and the pitch for each note is appended. A mapping function is created to convert string-based categorical data to integer-based numerical data.

(iii) Bayesian network is created using the Dirichlet Bayesian network score with an equivalent sample size of 1000.

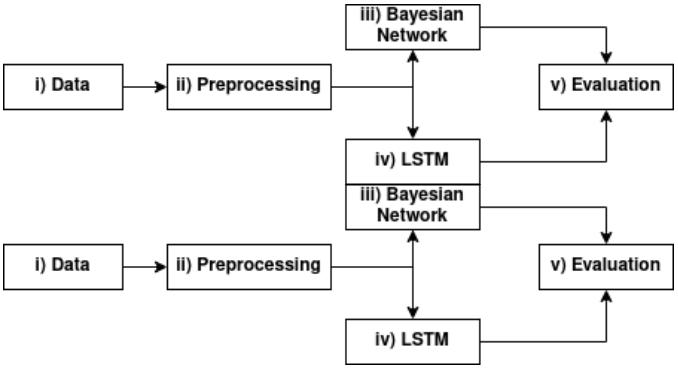

Fig. 1. An overview of the methodology in this paper.

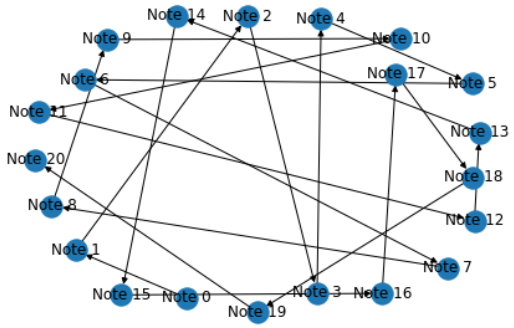

Fig. 2. Example of a Bayesian Network with a subset of 20 nodes representing positions of musical notes.

Bayes Parameter Estimation (BPE) is implemented to estimate the probability density function of the unknown parameters.

(iv) We use a one-column matrix as an input and a simple networking consisting of three LSTM layers, three Dropout layers, two Dense layers, and an activation layer (Softmax). In addition, we use cross-entropy to calculate loss and implement RMSprop optimiser to optimise the network.

(v) A Turing test is performed to evaluate the performance of the outputted music.

A. Preprocessing

To train the models, we first extracted the data from the MIDI files and converted the key signature of each MIDI file to the key of C major/A minor. The pitch and duration of each note are appended to its respective note. Chords are appended by encoding the ID of every note in the chord together into a single string. These encodings become useful when decoding the output of the models. In addition to this, the model views the same notes, played for different durations, as two separate entries (i.e. a C# played for 0.5 seconds will have a different value to a C# played for 1 second).

In this experiment, we assume a time step of 0.25 seconds and 8 notes per time step. This corresponds to a time signature of 4/4. The assumption allows for the calculation of rest notes by dividing the gap between notes by the selected time step and subtracting 1 (as to not double count). For example, if one note started at 0.5s, and the next note only begins at 1.0s, then one note is required to fill the gap.

B. Models

For this paper, we predefined the structure of the Bayesian network with 32 nodes. The current node is dependent on the preceding node. An example of the structure is shown in Fig. 2. We trained the Bayesian model to learn parameters of the network using the Bayes Parameter Estimation (BPE) and Dirichlet Bayesian network score with an equivalent sample size of 1000.

The structure used for the LSTM are as follows: 3 dropout layers, two dense layers, and an output layer the same size as the training set. This can be visualised in Fig. 3. With a dropout rate of 0.3, a cross-entropy loss function, the RMSprop optimizer, and 200 epochs, the LSTM was trained with an input size of 32 notes [5].

Once the Bayesian network and LSTM had finished training, they were provided with the first 32 notes randomly selected from the training data. They would predict a note, append the predicted note to the provided notes, and predict the next note, continuously shifting the given notes by one for their respective notes. This continued until 20 notes had been generated for each model. The notes were then converted into MIDI files, to be used for evaluation.

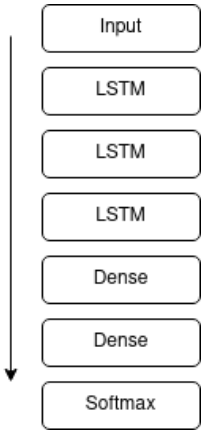

Fig. 3. Structure of the LSTM broken down into the 3 different layers.

C. Evaluation

We conducted a survey of people with no formal music education. The survey consisted of two parts. The first part involved playing songs made by humans and songs made the Bayesian network. Each participant was asked if the song was human-made or computer-generated. The second part involved a similar process, except with songs generated by the LSTM and human-made music. The same question was as in the first part.

D. Ethics Clearance

The study ethics application has been approved by the University's Human Research Ethics Committee (Non-Medical). The ethics application addresses key ethical issues of protecting the identity of the participants involved in the study and ensuring the security of data. The clearance certificate protocol number is *CSAM-2020-03*.

IV. Results

This section presents the performance of the Bayesian and LSTM networks in generating music, which sounds similar to that of human-made music. To evaluate the generated music, we conducted a survey where 20 participants were played both human-made music as well as machine-made music. The participants then had to determine which was human-made and machine-made.

Both models managed to learn and generate music which deceived the participants. In Fig. I, we see that the LSTM on average was identified as human-made 36% of the time, while the Bayesian network, on average, had been misidentified 39% of the time. In many cases, the participants required the songs to be repeated multiple times, as they could not immediately gauge which was real and which was not. Some went as far as to guess, due to not being 100% certain when given the choices.

TABLE I
PERCENTAGE OF PARTICIPANTS WHO INCORRECTLY IDENTIFIED THE MACHINE-MADE MUSIC.

Bayesian Network	LSTM
39.0%	36.0%

V. Discussion and Conclusion

The results indicate that the Bayesian network is a marginally more suitable model for the composition of music than the LSTM. These results suggest that there may be an underlying probabilistic dependency between the current note and the previous note. This may also indicate that music is more probabilistic than time-dependent as seen by the Bayesian network outperforming the LSTM in the Turing test.

The performance of both models displays a possibility of further application in the musical industry. With some refinement and alterations, the Bayesian network and LSTM may be used, for example, help artists who may struggle to come up with a melody for a song or even use the models to create accompaniment music.

However, there are limitations which may affect these conclusions. Due to the limited availability of hardware, the models had to be limited in complexity. We had to limit the number of songs used for training to reduce the amount of training time needed. This may have caused overfitting in the models. Another limitation seems to be with the survey itself. Some interviewees mentioned that as the survey progressed, they started to realise what was and was not machine-made music. This may be one of the reasons why the LSTM had worse results than the Bayesian network since the LSTM was evaluated by the interviewees after the Bayesian network.

This paper has presented a model which can not only generate music which fools a human but also outperform an LSTM in a Turing test. Future research may look at more complex variations of the Bayesian network as well as alternative parameters for the LSTM than the ones used in [5]. However, Bayesian networks have displayed potential in algorithmically composing music.

References

[1] S. A. Hedges, "Dice music in the eighteenth century," *Music & Letters*, vol. 59, no. 2, pp. 180–187, 1978.

[2] P. Langston, "Six techniques for algorithmic music composition," in *Proceedings of the International Computer Music Conference*, vol. 60. Citeseer, 1989.

[3] D. Eck and J. Schmidhube, "A first look at music composition using lstm recurrent neural networks," *Instituto Dalle Molle di studi sull' intelligenza artificiale*, 2002.

[4] C. Bell, "Algorithmic music composition using dynamic markov chains and genetic algorithms," *The Journal of Computing Sciences in Colleges*, vol. 27, pp. 99–107, 2011.

[5] P. Wiriyachaiporn, K. Chanasit, A. Suchato, P. Punyabukkana, and E. Chuangsuwanich, "Algorithmic music composition comparison," in *2018 15th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, 2018, pp. 1–6.

[6] D. Rizo, P. J. P. de León, C. Pérez-Sancho, A. Pertusa, and J. M. Iñesta, "A pattern recognition approach for melody track selection in midi files," in *Proc. of the 7th Int. Symp. on Music Information Retrieval ISMIR 2006*, 2006, pp. 61–66.

[7] B. Gavin, "What are wav and wave files and how do i open them?" 2018, https://www.howtogeek.com/392504/what-are-wav-and-wave-files-and-how-do-i-open-them, Last accessed on 2020-03-30.

[8] A. Bird, "Midi vs wav. what is the difference?" 2017, https://www.howtogeek.com/392504/what-are-wav-and-wave-files-and-how-do-i-open-them, Last accessed on 2020-03-30.

[9] A. Bhave, M. Sharma, and R. R. Janghel, "Music generation using deep learning," in *Soft Computing and Signal Processing*. Springer, 2019, pp. 203–211.

[10] I. S. Zhelavskaya, Y. Y. Shprits, and M. Spasojevic, "Chapter 12 - reconstruction of plasma electron density from satellite measurements via artificial neural networks," in *Machine Learning Techniques for Space Weather*. Elsevier, 2018, pp. 301 – 327.

[11] F. Colombo, A. Seeholzer, and W. Gerstner, "Deep artificial composer: A creative neural network model for automated melody generation," in *International Conference on Evolutionary and Biologically Inspired Music and Art*. Springer, 2017, pp. 81–96.

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, 12 1997.

[13] I. Ben-Gal, "Bayesian networks," *Encyclopedia of statistics in quality and reliability*, vol. 1, 2008.

[14] C. Raphael, "A bayesian network for real-time musical accompaniment," in *Advances in Neural Information Processing Systems*, 2002, pp. 1433–1439.

[15] T. Kitahara, *Music Generation Using Bayesian Networks*. Springer, Cham, 2017, pp. 368–372.

[16] A. Turing, "Computing Machinery and Intelligence," *Mind*, vol. 59, pp. 433–460, 1950.

[17] C. Ariza, "The interrogator as critic: The turing test and the evaluation of generative music systems," *Computer Music Journal*, vol. 33, pp. 48–70, 06 2009.

[18] A. Van Der Merwe and W. Schulze, "Music generation with markov models," *IEEE MultiMedia*, vol. 18, no. 3, pp. 78–85, 2011.

[19] A. Rodà, E. Schubert, G. De Poli, and S. Canazza, "Toward a musical turing test for automatic music performance," in *International symposium on computer music multidisciplinary research*, 2015.