Generating music using an LSTM network

Andrija Banić, Dario Pavlović, Marko Jurić, Antonio Babić, Mario Hladek
Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva, Zagreb, Hrvatska
{andrija.banic, dario.pavlovic, marko.juric, antonio.babic, mario.hladek}@fer.hr

***Abstract*—This study explores music generation using a Long Short-Term Memory (LSTM) neural network trained on the POP909 dataset with 250 songs. Data preparation involves MIDI file parsing, mapping to integers, and creating sequences for LSTM input. The model comprises three LSTM layers, batch normalization, and Dense layers. Hyperparameter experiments favor 10 epochs, a batch size of 1024, and a sequence size of 100. Music generation involves a sampling method and note generation process, combining learned patterns with controlled randomness. Comparative analysis with Google Magenta models ('Trained Magenta' and 'Pretrained Magenta') shows competitive performance. A structured survey assesses complexity, creativity, emotional impact, and overall enjoyment, revealing positive perceptions of the LSTM model's music generation capabilities.**

***Index Terms*—music generation,recurrent neural network, Long Short-Term Memory (LSTM), sequence creation, comparative analysis, deep learning, machine learning, human-computer interaction**

I. Introduction

The combination of artificial intelligence and music composition has witnessed significant advancements in recent years, offering novel ways for creative exploration. The history of this problem dates back to the 1950s, when Christopher Strachey wrote the first known algorithmic music program that produced a rendition of the British national anthem [1]. All the way to the modern Google magenta that was used to create models capable of generating melodies, harmonies, and even entire compositions. This paper dives into the realm of algorithmic music generation, employing a Long Short-Term Memory (LSTM) neural network trained on the publicly available POP909 dataset. As computational capabilities have evolved, using neural networks to analyze and generate intricate musical compositions has become a compelling area of research. The chosen dataset, POP909, provides a rich source of MIDI-formatted songs, capturing musical intricacies often associated with keyboard instruments. However, the sheer volume of data necessitated a selection of 250 songs for training the LSTM model, driven by considerations of computational efficiency. Our approach involves the application of LSTM, a type of recurrent neural network known for its ability to capture temporal dependencies. We dig into data preparation, model architecture, and hyperparameter optimization, seeking to get the LSTM's capacity to discern patterns and relationships within the musical domain. Beyond the technical details, this paper contributes to the broader discourse surrounding machine-generated music by comparing the performance of our LSTM model against two variants of Google Magenta models – 'Trained Magenta' and 'Pretrained Magenta.' This comparative analysis, paired with a structured survey evaluating subjective aspects of the generated music, tells us the artistic merit of our approach. In essence, this work aims to use a simple model to and compare its performance with some of the more advance models right now.

II. Similar worksheets

Beyond its conventional roles in tasks like prediction, classification, and translation, deep learning is gaining increased recognition as a method for music generation. This trend is evident in recent initiatives like Google's Magenta and Spotify's CTRL (Creator Technology Research Lab). The rationale behind this shift lies in leveraging the capabilities of deep learning architectures and training methodologies to autonomously grasp musical styles from diverse corpora and subsequently produce samples based on the inferred distribution. However, there are inherent challenges when directly applying deep learning to content generation, as it often leads to generated content merely mimicking the training set without showcasing genuine creativity. A comprehensive survey and analysis conducted by Briot et al. [2] delve into the landscape of deep learning techniques for generating musical content, shedding light on these challenges and the need for further exploration in achieving truly creative outputs.

While in the late 20th century, music generation predominantly relied on methods such as grammar-based [3] or rule-based systems [4], the contemporary approach has shifted towards machine learning, specifically deep learning. The motivation behind embracing deep learning, and machine learning techniques in general, lies in their inherent generality. This was demonstrated in the work of Fiebrink and Caramiaux [5], who shed light on the advantages of leveraging machine learning for creative music generation. They underscored two key benefits: firstly, the streamlining of the creation process in cases where the intended application is too complicated for analytical formulations or manual brute force design, and secondly, the notable robustness of learning algorithms compared to manually crafted rule sets. In their study, they found that there's a missing piece. Not many researchers have taken a close look at deep learning and artificial neural network techniques in the music world. They haven't thoroughly examined the problems, challenges, and solutions related to these advanced techniques. Most of the existing research is more focused on using these techniques for practical applications, like making apps. They emphasize the importance of conducting a detailed exploration to understand how deep learning can genuinely revolutionize the field of creating music.

Our choice of network architecture draws inspiration from a 2014 paper by Alex Graves [6]. In the paper, Graves demonstrates the effectiveness of Long Short-term Memory (LSTM) recurrent neural networks for generating complex sequences with long-range structure. In our context, we leverage Graves' insights to analyze and apply recurrent neural networks for sequence generation. Similar to Graves' methodology, our network predicts individual elements sequentially, allowing for the generation of intricate and realistic sequences.

III. Methodology

A. Dataset

For this project, we selected the publicly available POP909 dataset, comprising 909 songs in MIDI format. MIDI format represents songs as notes, commonly used for keyboard instruments. Alongside the MIDI songs, the dataset includes alternative versions of each song and additional unused data. Due to limited computing resources and high RAM usage, we opted to train our LSTM neural network model using a subset of 250 randomly chosen songs.

B. Data Preparation

We utilized the LSTM (Long Short-Term Memory) neural network for its superior memory retention, suitable for our music generation task. To prepare the dataset for model input, we developed the functions `get_notes_from_midi` and `prepare_sequences`. `Get_notes_from_midi` iterates through MIDI elements, extracting note pitches and chord information. The output is a list of strings representing individual notes (e.g., ”C4”) or chords (e.g., ”C4.E4.G4”). `Prepare_sequences` maps elements to integers, as LSTM requires numerical inputs. It creates fixed-length sequences from the list of notes, where each input sequence is paired with the next note. For instance, if the sequence length is 100, an input sequence might be [”C4”, ”E4”, ..., ”F4”], and the corresponding output would be the next note ”A4”. The sequences are then encoded to integer values and shaped to fit the model. This process took approximately one hour for 250 songs.

C. Model

We chose the LSTM neural network for its suitability in music generation. The model consists of three LSTM layers, each with 512 units and 30% recurrent dropout to prevent overfitting. Batch normalization follows the LSTM layers for stabilization. A Dense layer with 256 units interprets features learned by LSTM layers, and the final Dense layer has n_{vocab} units, representing unique notes and chords in the dataset. ReLU activation is used in the penultimate Dense layer, introducing non-linearity, and softmax activation in the final layer for multi-class classification. The model is compiled with categorical crossentropy loss and rsmprop optimizer.

D. Training

Experiments with epochs (5, 10, 20) indicated that 10 epochs were optimal in terms of time efficiency. Batch size was set to 1024, constrained by computational resources. Sequence size was chosen as 100, aligning with common practices in music generation. Using CUDA, the training process took a few hours.

E. Generating Music

After model creation and training, music generation is handled by two functions: `sample` and `generate_notes`. The `sample` method introduces randomness based on predicted probabilities, adjusted by a temperature parameter. `generate_notes` creates a sequence of music notes by selecting a starting point, generating notes in a loop, and using the `sample` method for controlled randomness.

IV. Results

In our comparative analysis, we employed two variants of Google Magenta models: 'Trained Magenta' using our dataset and 'Pretrained Magenta' pre-trained on Google Magenta's datasets. A structured survey evaluated model performance, with participants assessing original songs, our model's generated songs, and those from both Magenta variants. Criteria included complexity, creativity, emotional impact, and overall enjoyment. While our comparative analysis provides insights into the performance of different models in generating music, it is crucial to emphasize the inherent subjectivity of musical preferences. Unlike more quantifiable fields where accuracy or F1 scores might be employed for evaluation, music is a deeply personal and subjective experience. As such, there is no universal metric that definitively measures the success of a music generation model.

In figure 1., the first row, which featured original songs, demonstrated that respondents had a high ability to accurately identify them as human-composed. In the second row, showcasing our model's output, respondents perceived these songs as more reminiscent of human composition compared to the Trained Magenta model, and in one instance, even more so than Pretrained Magenta. The Trained Magenta's songs were predominantly identified as AI-generated. Meanwhile, Pretrained Magenta's compositions were largely recognized as AI-synthesized in the first set, but in the second set, it shifted towards them being perceived as crafted by a human.

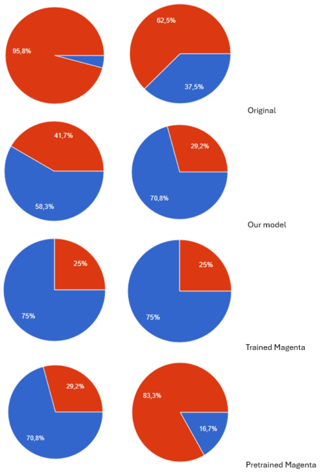

Fig. 1. Distribution of Responses to the Survey Question: ”Do you believe this song was generated by AI?” Red indicates a ”No” response, and blue signifies a ”Yes.”

After probing participants about their perceptions of AI-generated songs, they were prompted to assign ratings on a scale of 1-5 to each song for four distinct aspects. The average ratings were then calculated across all songs generated by a specific model, or in the case of the originals, for each of the aspects. Original songs received high ratings for complexity, creativity, emotion, and enjoyment, placing a strong emphasis on a robust emotional impact. Our model demonstrated a balanced performance across all four aspects. Google Magenta's model, trained on our dataset, exhibited heightened complexity and creativity but received lower scores for emotional impact and overall enjoyment. The pretrained Magenta model followed a similar pattern, showing a slight emotional and overall enjoyment improvement. An interesting observation is that both Magenta graphs(see Fig. 2 and Fig. 3) for average ratings of aspects appear inverted when compared to the original songs' average ratings, while our model seems to follow the original pattern. It's noteworthy that both Magenta models demonstrate higher complexity than the original and our model(see Fig. 4 and Fig. 5), with a slight edge in creativity. In short, respondents perceived original and our model's compositions as less complex but equally or more enjoyable and emotionally resonant than those produced by the Magenta models.

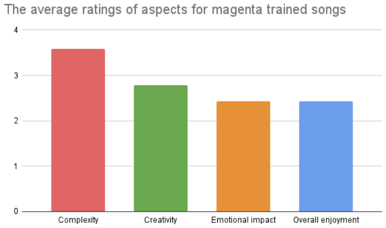

Fig. 2. The average ratings of aspects for magenta trained songs.

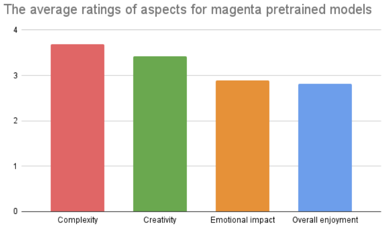

Fig. 3. The average ratings of aspects for magenta pretrained models.

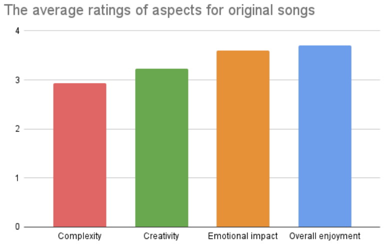

Fig. 4. The average ratings of aspects for original songs.

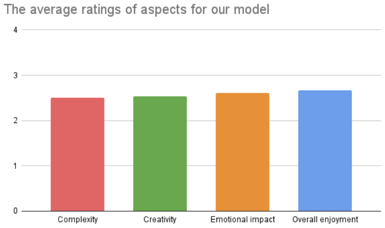

Fig. 5. The average ratings of aspects for our model.

V. Discussion

Our comparative analysis reveals interesting insights into the perceived quality of generated music across different models. Notably, both Magenta models exhibited higher complexity and creativity compared to the original songs and our model. However, participants generally perceived original and our model's compositions as less complex but equally or more enjoyable and emotionally resonant. This suggests that while Magenta models may excel in certain technical aspects, there is a preference among respondents for compositions that are simpler yet emotionally engaging.

VI. Conclusion

In conclusion, our study delves into the realm of algorithmic music generation, leveraging a Long Short-Term Memory (LSTM) neural network trained on the POP909 dataset. The model demonstrates competitive performance, with a focus on balanced complexity, creativity, emotional impact, and overall enjoyment. Comparative analysis with Google Magenta models provides valuable insights into the strengths and weaknesses of different approaches.

However, it is essential to acknowledge certain limitations in our study. The POP909 dataset, while rich in musical intricacies, comprises a total of 909 songs. For computational efficiency, we opted to train our LSTM model on a subset of 250 randomly chosen songs. This choice, driven by the computational demands of our approach, might impact the model's ability to capture the full diversity of musical styles and nuances present in the complete dataset.

Additionally, the computational demands for training the LSTM model, especially with larger datasets, can be significant. While our study provides promising results, further research with optimized computational resources and a more extensive dataset could offer deeper insights into the potential of AI in the realm of music composition.

References

[1] Crace, J., "First recording of computer-generated music – created by Alan Turing – restored," *The Guardian*, 2016, https://www.theguardian.com/science/2016/sep/26/first-recording-computer-generated-music-created-alan-turing-restored-enigma-code.

[2] Briot, JP. and Hadjeres, G. and Pachet, F., "Deep Learning Techniques for Music Generation," *Computational Synthesis and Creative Systems, Springer, London*, 2018, (Springer), https://arxiv.org/pdf/1709.01620.pdf.

[3] Steedman, M., "A Generative Grammar for Jazz Chord Sequences," *Music Perception*, 1984, https://homepages.inf.ed.ac.uk/steedman/papers/music/40285282.pdf.

[4] Ebicoglu,K. "An Expert System for Harmonizing Chorales in the Style of J.S. Bach," *The Journal of Logic Programming*, https://doi.org/10.1016/0743-1066(90)90055-A.

[5] Fiebrink, R. and Caramiaux, B., "The Machine Learning Algorithm as Creative Musical Tool," *CoRR*, Volume abs/1611.00379, 2016, http://arxiv.org/abs/1611.00379.

[6] Alex Graves, "Generating Sequences With Recurrent Neural Networks," *CoRR*, Volume abs/1308.0850, 2013, http://arxiv.org/abs/1308.0850.