J.N. Andrews Honors Program
Andrews University

Proposal for Senior Honors Thesis

HONS 497 Senior Honors Thesis **Credits** ___2___ (2 minimum required)

Directions: Please return signed proposal to the Honors Office **at least one week prior to your scheduled meeting with the Honors Council**. This proposal must be accepted by Honors Council the semester before presentation.

Student's Name: Nathaniel Patterson

Primary Advisor: Dr. Rodney Summerscales
Secondary Advisor:
Thesis Title: Finding an Embedding for Music Auto-Complete: An LSTM Approach
Local phone: (407) 712-3230 Email: nathanielp@andrews.edu

Expected date of Graduation: December 2020

I. Provide goals and brief description of your project or research.

Deep Neural Networks have been transformational to the field of Artificial Intelligence and Machine Learning. While a lot of the foundational theory and architectures were introduced in the late 20th century, it was not until 2006 that Neural Networks that the breakthroughs began that led to the current state-of-the-art (Liu, Weibo, et al, 2017). Neural Networks were transformational particularly in the field of Computer Vision by Convolutional Neural Networks (CNNs), as well as many Unsupervised Learning tasks. They have also seen success in their ability to model sequential data by Recurrent Neural Networks (RNNs). Many models have been created to handle the task of Text and Query Autocomplete, as well as Text Generation and have seen great success (Mirowski, 2015 & Pawade, 2017). In further research into sequence generation and completion, various RNN architectures have been applied to the problem of Music Generation with many seeing success in generating polyphonic music (Johnson, 2017). While much work and attention has been given to the problem of Music Generation, little to no work has been done on Music Autocomplete. This project seeks to introduce Music Autocomplete as a new problem, while adding to the body of knowledge on how Neural Networks process sequential data and how different data embeddings improve performance. This will also add to the subfields of music generation and the intersection of artistry and Artificial Intelligence.

The primary goal of my research is to analyze the effectiveness of two embedding strategies: notes as a string and notes as objects using a Long-Short Term Memory Recurrent Neural Network (LSTM-RNN) for music auto-completion when trained on the corpus of a single artist: Erik Satie. In order to measure the effectiveness of the model, I will restrict six songs at random from the training set. The model will be trained on the remaining 32 songs using K-fold cross-validation (K=4) to avoid overfitting. This is in hopes to study how well the network can capture style, as well as whether or not the artist in question is predictable by this simple

LSTM-RNN architecture. Success will be measured quantitatively by using the top 5 next suggested notes or chords and tracking whether or not the correct note in any one of the restricted nine compositions appeared in the predictions.

The LSTM-RNN model will be created using python and the Keras framework for building Neural Networks. This will be done in Google Collaboratory, a web-hosted Jupyter Notebook which has GPUs available for boosting training speeds. The data will be in the Musical Instrument Digital Interface (MIDI) file format, and I will use Music21, an open-source python library developed and maintained by Michael Cuthbert and Christopher Ariza. This library will assist in mining data from the .MID files as well as preparing the data for the network. Much of the focus of this research will be placed on the success of different embeddings of the music files. As there is no prior work on this specific problem, I will start with the text string representation of the data, taking a by-character approach which is basic to text auto completer and generation research. I then will compare those results to an embedding that better communicates the significance of a note and its octave as a single entity, which also make up chords.

II. **Outline your methodology. Please be specific**. How does this achieve your goals and how reliable is it?

As the primary focus of this project is on the embedding, much of the programming aspect is specific to mining relevant data from the MIDI files. Using the Parser function from the Music21 library, I will convert the MIDI files into Stream objects which are iterable and contain the tracks with the notes, chords, and their respective offsets. Offsets are Music21 data that tell where within a piece of music a note or chord object is played. While iterating through each Stream, I will mine the Pitch Class from each Note object and the Pitch Class Lists from each chord object, as well as their respective offsets. Once I have compiled a list of each song in the form of Note and Chord objects, I will iterate through each song again and combine notes and chords which are played on the same offset into one chord object. This is an important step, as with sheet music the stems of each note indicate the "polyphonic voice" of the particular measure. However, because of this distinction, when the files are parsed, each note and chord object is grouped together by polyphony rather than timestep. As the goal of this project is to train for sequence prediction, it is vital that the model understands when a note is being played in the course of the song. Since this is a limitation of the Music21 library, I have designed an algorithm to convert each Stream object to this format. A detailed graphic of the methodology will be developed for the final project.

Once the files are converted, the first embedding I will test will be based on previous work for text autocomplete tasks. When using RNNs for Natural Language Processing (NLP), models are given a vocabulary based on a text corpus on which it is trained. This vocabulary includes each of the lowercase letters of the alphabet and any numbers, special characters, or punctuation within the text. In this case, the Music21 naming conventions for notes are used with octave markings from "0-7" and notes from "A-G" with "–, #" to indicate flat or sharp notes. Each time step will be separated by a space, and at each there will either be a single note or chord played. The spaces between each timestep will be crucial to the recommended autocompletion after training. Once the songs have been translated from Stream objects into a text string of notes and chords, they must then be vectorized (one-hot encoded) to be fed into the neural network. This means that for each character in the string, there is a corresponding vector that has a location for each item in the vocabulary and it will fire a "1" for an instance of that character and a "0" elsewhere.

The network will consist of three LSTM layers with dropout and batch normalization. The input will use a sequence length of forty-five characters in the text string of notes and chords, and will train the weights based on predicting the forty-sixth character. This sequence length was chosen as some of the individual tracks of music are fairly short in length, and it will provide around two measures of the song to provide context to predict the next note.

Once the model has been trained, the same process for the six songs which were left out of the training corpus will be converted using the same algorithm to handle notes, chords, and offsets within the Stream objects. Then the pitches at each timestep will be converted into a text string, however, each song will be kept separate so that when given the 45 characters of a song string, the model can predict the next note or character. Because each timestep is separated by a space, if in the first 45 characters of the song, the next character is not after a space, it may predict an octave marking, a sharp or a flat, or even another note played at the same timestep to form a chord. Using all the predictions and their relative probabilistic certainties, each song will have the top five suggested next notes. The top five suggestions will be displayed with the true result below for comparison. The model will be tested first with this text-based approach to draw comparisons and benchmark against work more commonly done with text strings.

The other embedding will use the same network, but instead of using the generic text embedding, it will use a vocabulary of notes rather than characters. This way the model will learn the significance of "C#4" or "E2" as a unified object rather than an instance of a letter, a number, and a special character. This will give a larger vocabulary, but hopefully will better communicate the significance of notes within a measure. Notes will be split up at each time step and from each chord. It will then be encoded and the model will train on it using the same layers as before. Comparisons between the two embedding strategies will be evaluated on three other random samples of six songs from the total 38.

III. Explain in what sense your project is original, unique, or beyond normal senior expectations. How does it relate to current knowledge in the discipline?

This project is unique in that to my knowledge there is no previous research done on music autocomplete. There is a lot of work on music generation, text generation, and text/query autocomplete. This paper seeks to find a good embedding for this problem and references techniques and approaches used in prior work for text autocomplete and music generation. As there are no baseline metrics for evaluating success, this paper will take similar measures to text autocomplete for measuring model effectiveness, as well as seek to develop a good measure for model effectiveness for this task. Further, other papers which tackle music generation often do not restrict their training corpus to one artist with the hopes of capturing his or her style. This project begs the question of whether or not an artist can be predicted based on other works, while introducing music autocomplete as a new problem.

IV. Include a substantive annotated bibliography of similar or related work.

Liu, Weibo, et al. "A Survey of Deep Neural Network Architectures and Their Applications." Neurocomputing, vol. 234, 19 Apr. 2017, pp. 11–26., doi:10.1016/j.neucom.2016.12.038.

A comprehensive survey on the history of Deep Neural Networks and various applications. The sections on recurrent neural networks will be cited in the paper introduction.

Hochreiter, Sepp, and Jürgen Schmidhuber. "Long Short-Term Memory." *Neural Computation*, vol. 9, no. 8, 1997, pp. 1735–1780., doi:10.1162/neco.1997.9.8.1735.

Ground-breaking LSTM architecture for RNNs that deals with the "vanishing gradient" problem in which the gradient that is propagated back through the network either decays or grows exponentially. This paper was the solution to the problem originally reviewed by Hochreiter, and will be cited when explaining the LSTM model and why it is considered the basis for most sequential data processing tasks.

Choi, Keunwoo, et al. "Text-Based LSTM Networks for Automatic Music Composition." *ArXiv.org*, 18 Apr. 2016, arxiv.org/abs/1604.05358.

A comparison study on two separate embeddings for LISTM-RNNs using both character-based RNNs and word-based RNNs. These were trained on data in *band-in-a-box* format. This text-based approach was for music generation, a similar problem, so the results of this study are important to the autocomplete task. The RNNs in the music autocomplete project will be a character-based text RNN similar to this study and a note-based RNN, which will be a more accurate embedding than a word-based RNN.

Johnson, Daniel D. "Generating Polyphonic Music Using Tied Parallel Networks." Computational Intelligence in Music, Sound, Art and Design Lecture Notes in Computer Science, 2017, pp. 128–143., doi:10.1007/978-3-319-55750-2_9.

The state-of-the-art in music generation which has seen great success in generating polyphonic music. While my project is not focused on architecture, it does hope to communicate the significance of notes through the model by means of improving the embedding. This article is referenced as being the current state-of-the-art for music generation, a similar problem. The conclusions of this project help guide my framework for focusing on a note-based embedding.

Pawade, Dipti, et al. "Story Scrambler - Automatic Text Generation Using Word Level RNN-LSTM." *International Journal of Information Technology and Computer Science*, vol. 10, no. 6, 2018, pp. 44–53., doi:10.5815/ijitcs.2018.06.05.

A study done using LSTM-RNN for text-generation for stories. This article deals with a similar problem to music-autocomplete (text-generation) and helped guide the workflow for my research. Story Scrambler is an LSTM-RNN trained on text corpuses, similar to the text corpus of the music files in my research. The results of this study and the statistics on success with different parameters were useful in the network design for the music-autocomplete task.

Liang, Feynman, et al. " 18th International Society for Music Information Retrieval Conference." ISMIR, *Proceedings of the 18th ISMIR Conference, Suzhou, China, October 23-27, 2017*, 2017, pp. 449–456.

A study done in music generation using an LSTM framework of data that was restricted to one composer (Bach) and his Chorales. The hopes were to capture polyphonic style in the generated samples. The network called "BachBot" saw great success in musical discrimination testing with no significant difference in listeners distinguishing the real Bach from BachBot. This article will add to my research as it is one of the few cases of a restricted domain for training to one artist with the hopes of capturing style. My research aims to predict the next note in a song that the model was not trained on with the corpus restricted to the works of Erik Satie. The approaches in this study helped guide my thought processes in introducing the music autocomplete as a new problem.

Mirowski, Piotr, and Andreas Vlachos. "Dependency Recurrent Neural Language Models for Sentence Completion." *ArXiv.org*, 5 July 2015, arxiv.org/abs/1507.01193.

This study suggests dependency RNNs for sentence completion. While not using LSTMs, this article seeks to learn context from a text corpus and improve on sentence autocomplete on unseen data. The metrics and approaches taken in this article are important to this research as they involve sentence autocomplete which is a related problem. The comparisons between model performance on test sets set a good baseline for quantitative evaluation of this model. While the numbers are not as relevant, it is worth noting in my research how my project relates to results in similar works.

Cuthbert , Michael, and Christopher Ariza. "music21 Documentation." *music21 Documentation - music21 Documentation*, web.mit.edu/music21/doc/.

The website for all Music21 documentation which was referenced constantly for developing the algorithms to mine MIDI data.

V. Provide a statement of progress to date and list the research methods coursework completed.

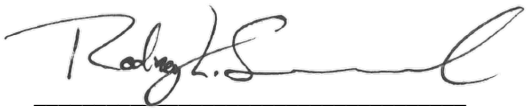
To date, the 38 pieces have been selected and downloaded to a GitHub repository. The model has been trained using the text string approach and saw good results for the simple model. This includes the aforementioned algorithms and code to sort and slice the data mined from the MIDI files, to transfer that data into a text string, to vectorize the text string, to build and train the model, and to generate predictions based on the unseen data. I am meeting semi-regularly with Dr. Max Keller professor of Music Theory and Composition, as this project percolates into some areas of music theory. Still to-do is code up the note-object approach, however most of the code from the note-string approach can be re-used for the note-object approach. The sampling will be done once the programs are written for both approaches.

Department Chair Approval

- **This student's performance in his/her major field is acceptable.**
- **He/she has completed the requisite research methods coursework for the research to be pursued.**
- **I understand that he/she plans to graduate with Honors.**

_____
Department Chair (signature)

Research Advisor Approval

I have read and support this proposal: 

Primary Advisor (signature)

I have read and support this proposal: William D. Wolfer

Secondary Advisor (signature)

If human subjects or if live vertebrate animals are involved, evidence of approval from the Institutional Review Board or an Animal Use Committee is needed through the campus scholarly research offices (Ext. 6361).