Title of your thesis

Your Name

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Your Department

Your Advisor, Chair
First Committee
Second Committee
Third Committee
Last Committee

December 4, 2020
Blacksburg, Virginia

Keywords: Some Keywords, Subject matter, etc.

Title of your thesis

Your Name

ABSTRACT

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Dedicated to Virginia Tech.

Acknowledgments

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

- Introduce the idea of musical research with computers. Talk about the illiac suite [15] and Music Information Retrieval.
- Significance of machine learning on the field
- Introduce idea of expressive musical performance. Brief conversation about the different performance components (articulation, dynamics, timing).
- Using Transformer architecture which hasn't been done in the field.
-

  Richard: report results

Chapter 2

Background

There are two major research components that this project is based on. The first is the problem domain of expressive musical performance (EMP), and the second is the ML modeling domain of Transformers. We will introduce both of these components and provide context for what makes them interesting as a research project and why they are worth exploring together. We start first with an overview and definition of EMP, and then a summary of the Transformer.

2.1 Expressive Musical Performance

EMP is a subset of the research field of Music Information Retrieval (MIR) [1] whose purpose is to use computational information to study, interpret, and gain a better understanding of the *essence* of music itself [17]. Perhaps the most well known MIR application is that of a musical recommendation system used by streaming services such as Spotify [2] to provide a personalized and unique experience for each user. However, as Widmer [17] suggests, there are a number of other non-trivial problems that face the field and will require significant effort from the research community to properly understand. A proper understanding of musical performance is one of them.

[1] Widmer [17] points out that MIR itself does not encompass the entire scope of computer music research, but that it is a good proxy to use when referring the field as a whole. We will operate under the same assumption

[2] spotify.com

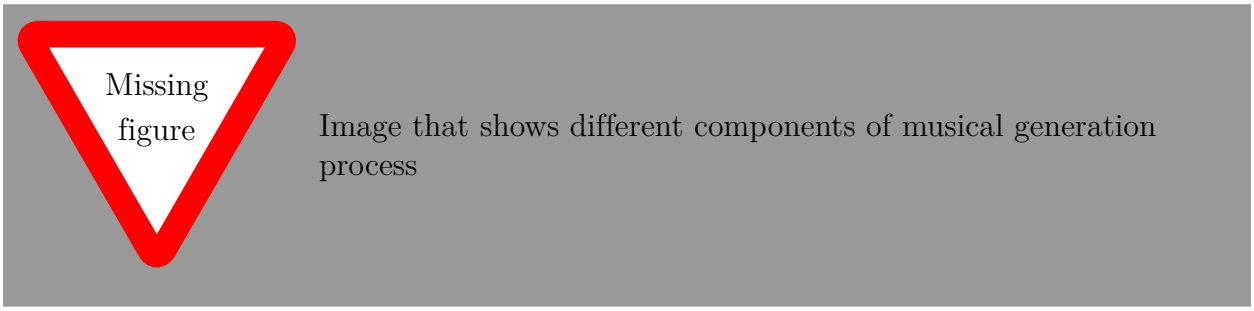

Figure 2.1: The first step of musical generation is composition, shown as a score in the figure. The second is performance, which is our area of interest. The third is the production of sound. Each different agent: composer, performer, instrument, and listener, can be thought of as a separate computational model in the generation process

MIR tasks can be broadly categorized in two ways - the first as computational methods for music analysis, and the second as computational methods for music generation. We are interested in the latter and its application in musical performance. In order to study how musical performance generation (and more particularly *expressive* musical performance generation) models work, it is necessary to gain a proper understanding of the entire computational musical generation process as a whole. Ji et al. [11] break the process down into 3 different components, with 4 different roles or agents that interact with that process. Figure 2.1 shows each step in the process as well as the agents that participate

Richard: Try to get permission to reproduce the image in the paper

.

An EMP model is analogous to the performer as show in 2.1, who takes as input a musical composition and produces as output a performance. It is the phenomena of musical expression that makes the performance generation process interesting. Musical expression can be thought of as the performers' interpretation of a composition codified into different performance parameters that are intended to increase the quality of the musical experience by the final listener. Because the quality of a musical experience is highly subjective, there

is no definition of what makes for a "correct" interpretation of a given composition [2]. The subjective nature of EMP generation makes it a difficult problem to understand from a computational perspective. However, it also makes it a highly intriguing research topic given that a clear understanding of the problem from a computational perspective will no doubt further our understanding of what exactly it is that makes music so subjective in the first place, and bring us one step closer to understanding music itself.

To properly understand exactly what it is that constitutes expression in musical performance, it is necessary to provide a detailed description of the first two components of the generation process - namely, scores and performances. We refer the reader to appendix A.1 which provides some basic terminology and concepts that will be useful for grasping the following section [3]. Due to the constraint of our data we focus only on western classical piano music.

Richard add reference

2.1.1 Scores

A musical score is a symbolic representation of a musical composition. The symbolic notation used to create musical scores can be thought of as a language used to express musical ideas and information. It presents this information in a hierarchical structure with different levels of musical detail at each level. The lowest level contains information about the pitch and timing of every single note, as well as optional information about how the note should be played. This can include information specific to instruments such as the bow direction of a violin, but for our purposes (dealing only with piano) we will consider this to be the articulation of each note, usually indicated by legato or staccato

[3] Most of the appendix material may seem elementary to those who already have a background in music or musical notation. However, we feel that is necessary to include if for no other reason than to provide a clear definition for our descriptions both in general and at detailed mathematical level

Richard: Make sure to have some background information on articulation in the appendix

The middle level contains information related to certain substructures within the musical composition, which are usually expressed within a grouping of notes or measures. The most common score annotations at this level are dynamic markings which indicate whether to play a grouping of notes loud (Forte), soft (Piano), or to gradually increase or decrease the volume (crescendo or decrescendo). Although dynamic markings are the most common at this level, it is also possible to see score markings for all other musical features, such as local tempo or articulation of a certain substructure. Perhaps the most important score marking at this level is that of a phrase, which is a marking that indicates that a group of notes should be interpreted as belonging to a singular musical idea and that each note should fit within the context of the phrase as a whole. A phrase can be expressed through all of the different aforementioned musical features, including the tempo, timing, dynamics, and articulation of the notes.

The highest level contains meta-information that relates to the entire composition as a whole. This information typically includes the key signature and time signature, as well as the global tempo for the entire piece, most commonly represented as BPM.

2.1.2 Peformance

An expressive musical performance contains most of the same musical information as does a score, but with one key difference; that is, that an expressive performance will deviate (or interpret) from the exact information that is presented in the score. For example, although a score may indicate a tempo of 120 BPM, it is highly unlikely that a given performer will perfectly adhere to this tempo throughout the entirety of the piece. This is even more

apparent if the score indicates a change in tempo somewhere in the composition. If a score indicates that the performance should speed up over a series of notes, there is no telling at what rate the tempo should increase. Some performers may choose to speed up at a fast rate and over a short period of time. Others may choose to increase the tempo at a slow rate and over a longer period of time. A single accelerando (a score indication to pick up the tempo) can result in either of these outcomes.

With that being said, a performance contains most of the same features related to a score, which include pitch, tempo, timing and articulation. Each of these expressive features will be measurable and absolute, whereas the score markings of these features can be viewed more as a suggestion than a rule. There a few additional features that are present in performances which are not in scores. The first we will refer to as deviation which is heavily related to timing. It is typically represented as a numerical number which represents how far off the timing of a particular note deviates from it's "correct" position in the score. These micro-timing deviations present in musical performances are an essential part of expression. Without them, indicating that each note onset and offset is exactly in line with its marking in the score, performances sound robotic and mundane .

Richard add a reference, graphic, and sample performance

The other important feature of performance that is not always present in a score applies specifically to the piano, and is the presence of a piano pedal. There are several different types of piano pedals, but the most common are the sustain pedal, which prolongs the duration of every note of the piano when activated, and the soft pedal which softens the sound of the entire piano. Although the effects of these pedals are directly related to the articulation and dynamics of the performance, their presence (or lack of) can be seen as a crucial component of piano performance. It is common for the sustain pedal to see active use in almost all modern piano performance, even when there doesn't exist any score marking indicating it's use.

Richard: Add section and reference to the specifics of feature engineering related to both the score and the performance in the methods section

.

2.1.3 Data

The data required for EMP generation includes some digital form of representation of a score as well as a corresponding performance. Scores are typically given in the form of MusicXML, which is a text-based representation of a score. Performances could be directly be rendered as audio which is the process used by human performers with the use of an acoustic instrument. Instead of audio however, an intermediate data form, MIDI, is used to represent the performance. This better aligns with the generation process outlined in 2.1. In the full generation process, a separate model would be used to take the performance data in MIDI and synthesize that into raw audio which would be presented to the listener. Both data formats contain all of the required information to represent all of the musical components of both a score and a performance, including pitch, tempo, timing, articulation, deviation, and pedal. See appendix A.2 for more information on both MusicXML and MIDI.

To build an EMP generation model, it is necessary to run both the score and performance through a data alignment process in which every note of the performance is mapped to it's corresponding position in the score. Given the highly dynamic nature of musical performance, it is a non-trivial task to run this alignment process for a set of scores and performances, especially if the task is performed by manual human annotation. There exist methods for both manual and automatic alignment. Due to the time-consuming nature of manual alignment and the need for large data sets to build higher quality models, automatic alignment algorithms are an active area of research.

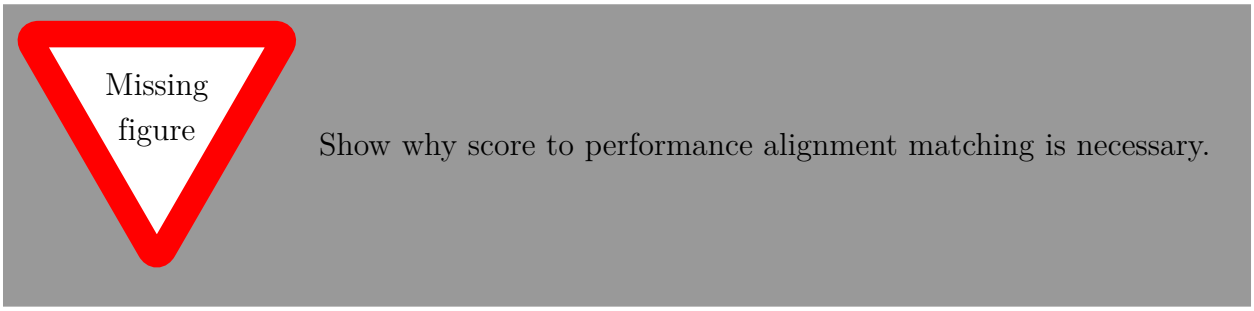

Figure 2.2: Two performances of the same score can vary wildly in their tempo and timing. This makes it necessary to have a score to performance alignment for every performance.

Richard: Add reference to section which gives relevant research

2.1.4 Performance Evaluation

One of the most important components of any computational model performing a task is that of evaluation. Evaluation is used to determine the quality of a model, and serves as a benchmark to compare different models used in the same task. Due to the inherently subjective nature of music and musical performance discussed in 2.1, evaluation is notioursly difficult to understand and perform correctly for EMP generation models [2].

Evaluation for computational models, specifically for EMP models, is typically categorized in two ways, quantitative evaluation and qualitatiative evaluation. Quantitative evaluation methods involve using numerical metrics which are computationally generated and deterministic. Qualitiative evaluation methods usually involve some form of human feedback and judgement presented in some standardized statistical measures. The key difference between quantitative and qualitiative is that qualitative methods are not as consisten and much more difficult to reproduce, given the reliance on the subjective feedback of human listeners. Traditionally, quantitative methods are preferred because of their consistency and reliability, In the case of EMP models however, qualitative evaluation methods may be even more impor-

tant in gaining an understanding of what makes one model better than another. Finding good methods of evaluation is an active area of research in EMP [2].

Quantitative

This method of evaluation is standard for ML models in general. There are a number of different metrics which are used in the evaluation process, all of which are specific to type of data and problem domain the model fits inside of. We will briefly cover the most common quantiative evaluation method that applies to our data and modeling domain, which is regression .

Richard

Add reference that discusses the feature engineering

The two common metrics used for evaluation and regression are Mean-Squared-Error (MSE) and the Pearson Correlation Coefficient, usually denoted as the R^2 score. MSE is used to measure the difference between a prediction and an actual observed target value, and can be denoted as $MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$, where Y_i is the observed value at time step i, and $\hat{Y}_1$ is the predicted value. R^2 is a probablistic measure of the linear correlation between variables X and Y, and is denoted as $\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y}$ where cov indicates the covariance and σ indicates the standard deviation. [4]

One of the problems with using quantiative, or "objective" evaluation methods, is that it usually invovles comparing a generated performance $\hat{Y}$ with a human performance Y. Given that no performance (or interpretation) of a can objectively been seen as better than another, this method of evaluation is also biasing the quality of a model towards some subjective view of the "correct" interpretation. Of course, a "correct" interpretation doesn't exist, which is what makes evalution methods for this particularly problem difficult.

[4] See wikipedia for more information on MSE, covariance, standard deviation, and the correlation coefficient

Qualitative

Richard: Need to conduct more research before I can write this section. Haven't done so because I won't be performing a qualitative evaluation myself in the paper. However it is still worth mentioning

2.2 Transformers

To properly understand the significance of Transformers and their involvement in our work, it is necessary to provide context about the domain in which the Transformer was first introduced and give an overview of the existing work in that domain that the Transformer built on. We'll then provide some detail about the Transformer itself as well as adaptations of the original architectures and their results.

2.2.1 Natural Language Processing and Machine Translation

One of the most commonly studied fields in Machine Learning and Artificial Intelligence is Natural Language Process (NLP), which (similarly to MIR) uses computation to ascertain a better understanding of human language as well as build technological tools that are useful in performing common language tasks. One such task is that of machine translation, which involves using computation alone to translate text from one language to another. NLP research usually invovles building sequence-based models (which explore the individual elements of an ordered set of items) due to the inherently sequential nature of language, as opposed to a non-sequential model which doesn't account for sequential data, such as a single image. Machine translation falls under the category of sequence-to-sequence (seq-2-seq) modeling problems, which involve the mapping and relationship of one sequence to

another. This is typically in the form of translating a single sentence from one language (English) to another (French).

More specifically, machine translation (and other seq-2-seq tasks) can be defined as taking an input sequence $\mathbf{x} = \{x_1, x_2, x_3, ...x_m\}$ of size m and producing an output sequence $\mathbf{y} = \{y_1, y_2, y_3, ...y_n\}$ of size n such that $M(\mathbf{x}) = \mathbf{y}$, where M can be any machine translation model. In some seq-2-seq tasks, $m = n$ are the same, implying that the input and output sequence are the same length. As is often the case in language translation, the input sentence and output sentence are of varying lengths, so we can assume that $m \neq n$.

It is common to use an encoder-decoder architecture for M, where there is an encoder E which takes in the input data and outputs and finds some hidden representation $E(\mathbf{x} = \mathbf{z})$. This hidden representation is given as input to the decoder, and the decoder uses it to produce the final output, $D(\mathbf{z}) = \mathbf{y}$. We can then define an encoder-decoder seq-2-seq model as $M(x) = D(E(\mathbf{x})) = \mathbf{y}$. Historically, a Long-Short-Term-Memory neural network (LSTM)[5] has been used for both E and D, where the hidden representation $\mathbf{z}$ has been a fixed length vector .

Richard

Add reference

One of the limitations of such a model is that it has to compress all of the information of the input data into the fixed-length vector $\mathbf{z}$ which causes the network to potentially lose important information, particularly in the case where an input sentence is given to the network which is longer than any present in the training data. Bahdanau et al. [1] present the attention mechanism which, used in conjunction with an RNN based encoder, allows for the hidden representation to itself be a sequence $\mathbf{z} = \{z_1, z_2, z_3, ..., z_m\}$ of size m (the same size as the input sequence). Each z_i element in the sequence contains information about the whole input sequence, with an emphasis on the elements closest to the i-th element.

[5] An LSTM is a common variant of a Recurrent Neural Network (RNN) which is the most standard deep learning model used for sequence modeling. See https://en.wikipedia.org/wiki/Long_short-term_memory

This allows the hidden representation to encode any relationship that one element in the sequence has with another. The decoder then uses this information to ”pay attention” to words in the output sequence that have a relationship with words in the input sequence, given the context that is encoded in the hidden representation at a particular time step i. The attention mechanism and model that uses it achieved state of the art results in the machine translation task, due in part to the fact that hidden representation is not limited to a fixed-size vector. The original attention mechanism presented by Bahdanau et al. [1] and its adaptations have since been used in tandum with recurrent models to improve the state of the art in several sequence modeling tasks . One of the limitations with standard recurrent networks is their inability to retain information across long sequences - attention provides a way to create additional context and better memory across these longer sequences which has led to the increase of performance in attention-based models. .

Richard find reference

Richard find reference

2.2.2 Attention is All You Need

In the seminal paper, Vaswani et al. [16] introduce the Transformer. The Transformer is an encoder-decoder seq-2-seq modeling neural network architecture that relies solely on the use of attention and cuts out any semblance of a recurrent architecture. The Transformer was the first architecture to make use of attention by itself, and by doing so pushed the state of the art in machine translation even further than it had been with attention-based recurrent models.

The Transformer architecture consists of a stack of N layers, all of which use a combination of a self-attention (attention that applies only within a single input sequence and not between an input and a output sequence)

Richard: Explore different ways to describe self-attention. May not even be necessary at all to mention

mechanism along with a standard pointwise fully connected feed-forward neural network (FFNN). Both the encoder and decoder comprise of these attention based stacked layers. For a full description of the architecture see [16].

2.2.3 Transformer Adaptations: BERT and GPT

Of particular interest in the new Transformer modeling domain is powerful adaptations of the original architecture which have been applied to many other NLP tasks besides machine translation. On such architecture, BERT (which stands for Bidirectional Encoder Representations from Transformers), uses what can be referred to as an "encoder only" Transformer model.

The original Transformer was built with machine translation in mind, but there are several other NLP tasks that could possibly benefti from using an attention only architecture. Some of these tasks include standard text classification, textual entailment, sentiment analysis, question answering, and many more . BERT was introduced as an encoder only transformer model that could generalize to all of these tasks. The method which it made use of was pre-training the model on a massive data set, with the intuition that by feeding the model so much data that it would learn a general representation of language that could then be applied to several different tasks. BERT is effectively a massive encoder for language in general, and can be used in conjuction with other models as simple decoders to perform these tasks. See the original paper[3] for the full architecture and details.

Richard Find reference

Similarly to BERT, the Generative Pre-trained Transformer (GPT) architecture[14] is an adaptation of the original Transformer. The GPT architecture can be seen as a "decoder

only" transformer, and is used as a general Language Model (LM). The task of a LM is simple; to predict the next word in a sequence of given words. Given that GPT is a generative model, it employes the decoder side of the Transformer, which is responsible for actually generating the text as part of the machine translation taks. Similarly to BERT, GPT models are pre-trained on massive amounts of data to learn a general representation of language, and used in conjuction with other models to perform various tasks.

Both BERT and GPT have significatly pushed the state of art in NLP and sequence modeling in general. Their success in the domain of language presents questions about their effectiveness in other related domains, such as music.

Chapter 3

Related Work

3.1 Existing EPG models

- KTH system [6]. A rule-based system for expressive performance. Rules are selected through a empirical process based on human feedback.
- YQX. A Bayesian network that models timing, dynamics, and articulation [18]. Won the 2008 RenCon contest.
- Basis Function Models [4]
  - Linear Basis Functions. Uses Least Squares regression and Bayesian models with about the same performance
  - Non-Linear Basis Functions. Uses standard feed-forward network. FFNN perform better than Linear models. Also uses an RNN.
- Giraldo and Ramirez use several different ML algorithms, including Decision Trees, k-NN, SVM's, and FFNN to build an expressive performance generation system for improvisational Jazz guitar [7].
- Moulieras and Pachet use a Maximum Entropy model to infer the underlying distribution of expressive performance and build a generation system trained from a mix of popular music. Their expressive model outperforms base models in listening tests [13].

Richard

This needs more exploration. Lot of possibilities for future work

- Jeong builds two versions of virtuosoNet, one using a recurrent hierarchical attention network (HAN) [9], and another using a recurrent graph network [10]. These models are built using a dataset order of magnitudes larger than other datasets and attempt to model the expressive performance feature of the pedal, which no other model does. The code for the models is also open source so it was chosen as the starting place for this work.

  Richard: Add more papers and expand upon the existing research a bit more. Isn't completely necessary but will be good for my overall understanding

3.2 Datasets

- Talk about the fundamental limitations of gathering data for this problem, especially in relation to other fields [6]. Because of this, the lack of high-quality data is limited.

- The dataset used for the virtuosoNet [10] [9] will be the dataset used for the experiments. At the time the experiment started it was the largest publicly available dataset applicable to the EPG systems, and was chosen for use. A recent publication [5] builds off of the dataset used for the virtuosoNet with more sophisticated alignment and some extensions to the size (dataset is named ASAP). ASAP would be more appropriate for future use.

- One of the necessary data processing tasks for EMP is the alignment between the score and performance of a given piece. Because there is always an inherent interpretation of a composition by a performer Richard: Reference this in the introduction, there is no clear mapping between any given score and performance. It remains necessary to have some sort of alignment process to match each note in the performance with its related position in the score.

Richard: This needs more research. Find relevant papers to cite, as well as show a diagram that makes it clear why alignment is necessary

.

Chapter 4

Experiments

4.1 Model and Experiments

- Due to the open-source nature of virtuosoNet project and its attempt to build a more cohesive EPG model by introducing the pedal as an expressive feature and training on a much larger dataset, we built off of this model.

- Because of the significant advances in other sequence modeling domains (such as NLP) and the indication of increased performance of another related task with the Music Transformer [8], the main question we want to answer is whether we can see similar increases in model performance by applying a Transformer ANN architecture to the problem domain.

- We will experiment with a transformer encoder only architecture similar to BERT. The problem includes a 1-1 to mapping between every note in the score and a related note in a performance. This is different than seq-2-seq modeling problem such as neural machine translation which maps a sequence of one length to another sequence of a different length, which is what the full Transformer architecture was intended for. The Transformer Encoder can be seen as as a large encoder that learns the best representation for a given feature set. The model we'll build will use a simple FFNN that accepts the output of the transformer encoder to decode this representation and

give the final feature set which is then used to create a performance. This is similar to the BERT architecture and it's intended application.

Richard: Come up with a more detailed explanation of this modeling choice. Also create a visual diagram that explains the transformer encoder with the simple regression model sitting on top of it

- Because we are using the same dataset used to train virtuosoNet, we will directly compare the performance a Transformer model to the existing virtuosoNet models using the same quantitative metric, MSE.

Richard: Come up with specific model experiments and comparison in a table. Table doesn't have to have results but needs the general outline that will be used in the final paper

4.2 Evaluation

- Quantitative: Because we are using the same dataset used to train virtuosoNet, we will directly compare the performance a Transformer model to the existing virtuosoNet models using the same quantitative metric, MSE.

Richard: Come up with specific model experiments and comparison in a table. The table doesn't have to have results but needs the general outline that will be used in the final paper

- Due to time and resource constraints, no sophisticated qualitative evaluation was conducted for the models. However, a personal evaluation was used during the entire model development process.

Richard: Talk about method used for personal analysis

-

Chapter 5

Results

5.1 Quantitative

Add table with results of experiments along with explanations.

5.2 Qualitative

Give personal qualitative report.

Chapter 6

Discussion

Richard: The following are some interesting discussion ideas that have come up so far. There is no telling if these will be in the final paper or not after conducting more experiments.

- Transformer performs worse according the quantitative metrics. This could be because it doesn't build in a specific hierarchical layer that is specific the problem. It is a much more generic model. There is a lot of room for exploration into experimenting with different architectures based on the Transformer to better fit the problem domain.

  Richard: Add more discussion based on more results

- Transformer appears to be a more dynamic model than the recurrent virtuosoNet model that makes more "mistakes". Does this mean that it is more "human".

Richard

Add discussion of uncanny valley

- Pedal in performance is messy. Could be because of problems in the feature and modeling, or could just be because it is a difficult problem to model.

-

  Richard: Discussion on qualitative results

Appendices

Appendix A

Appendices I

A.1 Musical Concepts and Terminology

A.1.1 Pitch

The first and most basic component in music is pitch. Pitch is a perceptual property of sounds that relates to the physical frequency of a sound vibration [12]. It is what determines whether or not a sound can be though of as ”high” or low”. The most commonly known way to conceptualize pitch is the 88 different keys on a piano keyboard, where each key represents a different patch value. Pitch is most commonly labeled using scientific pitch notation, which couples a range of letters (A to G) with a range of numbers (zero to eight) that correspond to different octave ranges [1]. The most well known pitch is C4, or ”middle C”, and lays in the very center of a standard 88 key piano.

Richard

Create or find visual-ization

A.1.2 Tempo and Timing

Tempo in music describes the rate at which notes are played, and timing describes when a particular note should be played relative to the start of the composition. They are best explained in the context of modern western musical notation introduces the idea of note

[1] https://en.wikipedia.org/wiki/Scientific_pitch_notation

durations, time signatures, measures, and beats [2].

Richard: Find a more intuitive way to explain this. The piano roll explanation and visualization may work better

Each composition is broken down into a sequence of measures, and the time signature defines how many beat exist per measure, as well as the duration of a single beat. For example, a 4/4 time signature indicates that there are 4 beats per measure (the top half of the time signature), and that the duration of each beat is represented by a quarter note. A 3/4 time signature would indicate only 3 beats per measure, with the beat duration represented by a quarter note. The timing of a note would refer to it's measure, beat, and note duration. Tempo is most commonly given in beats per minute (BPM). A composition with a 4/4 signatue and a 120 BPM would mean that after one minute, 30 measures of the composition should have been played so far.

Richard create or find visualization

A.1.3 Dynamics

Dynamics can simply be thought of as how loud or soft a note should be played (or has been played).

A.2 Data Representation

A.2.1 MusicXML

A.2.2 MIDI

[2] See https://en.wikipedia.org/wiki/Musical_notation#Modern_staff_notation for a more detailed explanation

Bibliography

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[2] Carlos E Cancino-Chacón, Maarten Grachten, Werner Goebl, and Gerhard Widmer. Computational models of expressive music performance: A comprehensive and critical review. *Frontiers in Digital Humanities*, 5:25, 2018.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Carlos Eduardo. *Computational modeling of expressive music performance with linear and non-linear basis function models*. PhD thesis, JOHANNES KEPLER UNIVERSITY LINZ, 2018.

[5] Francesco Foscarin, Andrew Mcleod, Philippe Rigaux, Florent Jacquemard, and Masahiko Sakai. Asap: a dataset of aligned scores and performances for piano transcription. In *ISMIR 2020-21st International Society for Music Information Retrieval*, 2020.

[6] Anders Friberg, Roberto Bresin, and Johan Sundberg. Overview of the kth rule system for musical performance. *Advances in Cognitive Psychology*, 2(2-3):145–161, 2006.

[7] Sergio Giraldo and Rafael Ramirez. A machine learning approach to ornamentation modeling and synthesis in jazz guitar. *Journal of Mathematics and Music*, 10(2): 107–126, 2016.

[8] Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Noam Shazeer, Ian Simon, Curtis Hawthorne, Andrew M Dai, Matthew D Hoffman, Monica Dinculescu, and Douglas Eck. Music transformer. *arXiv preprint arXiv:1809.04281*, 2018.

[9] Dasaem Jeong, Taegyun Kwon, Yoojin Kim, Kyogu Lee, and Juhan Nam. Virtuosonet: A hierarchical rnn-based system for modeling expressive piano performance. In *ISMIR*, pages 908–915, 2019.

[10] Dasaem Jeong, Taegyun Kwon, Yoojin Kim, and Juhan Nam. Graph neural network for music score data and modeling expressive piano performance. In *International Conference on Machine Learning*, pages 3060–3070, 2019.

[11] Shulei Ji, Jing Luo, and Xinyu Yang. A comprehensive survey on deep music generation: Multi-level representations, algorithms, evaluations, and future directions. *arXiv preprint arXiv:2011.06801*, 2020.

[12] Anssi Klapuri. Introduction to music transcription. In *Signal processing methods for music transcription*, pages 3–20. Springer, 2006.

[13] Simon Moulieras and Francois Pachet. Maximum entropy models for generation of expressive music. *arXiv preprint arXiv:1610.03606*, 2016.

[14] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[15] Orjan Sandred, Mikael Laurson, and Mika Kuuskankare. Revisiting the illiac suite–a rule-based approach to stochastic processes. *Sonic Ideas/Ideas Sonicas*, 2:42–46, 2009.

[16] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30:5998–6008, 2017.

[17] Gerhard Widmer. Getting closer to the essence of music: The con espressione manifesto. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(2):1–13, 2016.

[18] Gerhard Widmer, Sebastian Flossmann, and Maarten Grachten. Yqx plays chopin. *AI magazine*, 30(3):35–35, 2009.