Creating Musical Scores Inspired by the Intersection of Human Speech and Music Through Model-Based Cross Synthesis

William Alexander Thompson IV

Louisiana State University and Agricultural and Mechanical College

Follow this and additional works at: https://repository.lsu.edu/gradschool_dissertations

Recommended Citation

https://repository.lsu.edu/gradschool_dissertations/5876

This Dissertation is brought to you for free and open access by the Graduate School at LSU Scholarly Repository. It has been accepted for inclusion in LSU Doctoral Dissertations by an authorized graduate school editor of LSU Scholarly Repository. For more information, please contact gradetd@lsu.edu.
CREATING MUSICAL SCORES INSPIRED BY THE INTERSECTION OF HUMAN SPEECH AND MUSIC THROUGH MODEL BASED CROSS SYNTHESIS

A Dissertation

Submitted to the Graduate Faculty of the Louisiana State University and Agricultural and Mechanical College in partial fulfillment of the requirements for the degree of Doctor of Philosophy

in

The School of Music

by

William Alexander Thompson IV
B.M., University of New Orleans, 2006
M.M, University of New Orleans, 2012
August 2022
Acknowledgments

I would like to thank the faculty and student of the LSU Experimental Music and Digital Media program for inspiring me in ways I had not known were possible. Jesse Allison, Mara Gibson, Stephen Beck and Edgar Berdahl have all been very supportive and giving. Thank you. I’d also like to thank my family, Eva, Bill and Matthew for years of encouragement. I would like thank my wife Erin for her wisdom and love. Finally, I’d like to thank my maker for the beauty of this world and beyond.
Table of Contents

ACKNOWLEDGMENTS ................................................................. ii
LIST OF FIGURES ................................................................. v
ABSTRACT .................................................................................. vii

CHAPTERS

1 INTRODUCTION ................................................................. 1

2 RELATIONSHIPS BETWEEN SPEECH AND MUSIC .......... 4
   2.1 Origins of Music in Relation to Speech ....................... 4
   2.2 Rhythm in Speech ...................................................... 6
   2.3 Pitch in Speech ......................................................... 8
   2.4 Intervalic Relationships in Speech ............................. 9
   2.5 The Musical Tendencies of Emotional States of Speech .... 10

3 HISTORICAL CONTEXT ..................................................... 14
   3.1 Speech in Musical Composition ................................. 14
   3.2 Algorithmic Symbolic Musical Composition ................. 17

4 MAGENTA: GENERATING AND TRAINING WITH
   RECURRENT NEURAL NETWORKS ...................................... 20
   4.1 Magenta .................................................................. 20
   4.2 Recurrent Neural Networks ...................................... 22
   4.3 Magenta’s Polyphony RNN Generation Algorithm .......... 25
   4.4 Training ................................................................. 29

5 MODELS CREATED REPRESENTING TRADITIONAL
   MUSICAL SCORES .......................................................... 32
   5.1 Preparing Music Corpora For Training ...................... 32
   5.2 Individual Model Training and Evaluation of Model Performance ...... 34

6 MODELS CREATED REPRESENTING CHARACTERISTICS OF SPEECH
   ................................................................. 49
   6.1 Preparing Speech Corpora For Training ...................... 49
   6.2 Individual Model Training and Evaluation of Model Performance ...... 51

7 THE COMPOSITIONAL PROCESS THROUGH COMPUTATIONAL STEERING
   ................................................................. 59
   7.1 Cross Synthesizing Models ...................................... 59
   7.2 Music Inspired by Speech and Speech Inspired by Music .......... 59
   7.3 Successive Primers .................................................... 65
   7.4 Seven Piano Etudes Speaks the Moody Machine, Compositional Observations .......... 70
8 CONCLUSIONS ........................................................................................................................... 84

APPENDIX
A SEVEN PIANO ETUDES SPEAKS THE MOODY MACHINE,
FULL SCORE................................................................................................................................. 87
A.1 Etude I. 1440 Type (for human performer) ................................................................. 88
A.2 Etude II. 2029 Claim of Consciousness (for machine performer) ......... 91
A.3 Etude III. 1712 Steam Engine (for human performer) ..................... 108
A.4 Etude IV. 2030 Mind Upload (for machine performer) ...................... 110
A.5 Etude V. 1844 Telegraph (for human performer) ....................................... 120
A.6 Etude VI. 2040 Full Immersion (for machine performer) ................ 124
A.7 Etude VII. 2045 The Singularity (for human performer) ..................... 136
A.8 Etude VII. 2045 The Singularity (for machine performer) ................ 139
B CODE CORRESPONDING WITH ETUDE GENERATION .................. 146
B.1 Etude I. 1440 Type (for human performer) ................................................................. 146
B.2 Etude II. 2029 Claim of Consciousness (for machine performer) .... 147
B.3 Etude III. 1712 Steam Engine (for human performer) ............................. 148
B.4 Etude IV. 2030 Mind Upload (for machine performer) ...................... 149
B.5 Etude V. 1844 Telegraph (for human performer) ........................................ 150
B.6 Etude VI. 2040 Full Immersion (for machine performer) ................. 151
B.7 Etude VII. 2045 The Singularity (for human and machine performer) ........................................... 152

REFERENCES ............................................................................................................................... 153

VITA ............................................................................................................................................... 157
**List of Figures**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>MIDI models trained on music and speech</td>
<td>3</td>
</tr>
<tr>
<td>4.1</td>
<td>RNN Vectors</td>
<td>23</td>
</tr>
<tr>
<td>4.2</td>
<td>LSTM Cell</td>
<td>25</td>
</tr>
<tr>
<td>5.1</td>
<td>Preparing the Data-Set for Model Training</td>
<td>33</td>
</tr>
<tr>
<td>5.2</td>
<td>D(u)o published score example</td>
<td>36</td>
</tr>
<tr>
<td>5.3</td>
<td>D(u)o MIDI corpus example</td>
<td>37</td>
</tr>
<tr>
<td>5.4</td>
<td>D(u)o generation example</td>
<td>38</td>
</tr>
<tr>
<td>5.5</td>
<td>Thelonious Monk Corpus Example</td>
<td>40</td>
</tr>
<tr>
<td>5.6</td>
<td>Thelonious Monk Generation Example</td>
<td>41</td>
</tr>
<tr>
<td>5.7</td>
<td>James Booker Corpus Example</td>
<td>43</td>
</tr>
<tr>
<td>5.8</td>
<td>James Booker Generation Example</td>
<td>44</td>
</tr>
<tr>
<td>5.9</td>
<td>William Thompson Corpus Example</td>
<td>46</td>
</tr>
<tr>
<td>5.10</td>
<td>William Thompson Generation Example</td>
<td>47</td>
</tr>
<tr>
<td>6.1</td>
<td>Preparing the Data-Set for Model Training</td>
<td>51</td>
</tr>
<tr>
<td>6.2</td>
<td>Vokaturi evaluation of Neutral Speech Audio Dataset</td>
<td>54</td>
</tr>
<tr>
<td>6.3</td>
<td>Vokaturi evaluation of Neutral Speech Model Generation for Machine and Human Performance</td>
<td>54</td>
</tr>
<tr>
<td>6.4</td>
<td>Vokaturi evaluation of Amused Speech Audio Dataset, Corpus for Machine Performance, Corpus for Human Performance</td>
<td>55</td>
</tr>
<tr>
<td>6.5</td>
<td>Vokaturi evaluation of Amused Speech Model Generation for Machine and Human Performance</td>
<td>56</td>
</tr>
</tbody>
</table>
6.6 Vokaturi evaluation of Angry Speech Audio Dataset, Corpus for Machine Performance, Corpus for Human Performance .......................... 56

6.7 Vokaturi evaluation of Angry Speech MIDI Corpus for Machine Performance ................................................................. 57

7.1 Speech Primer with Musical Model ................................................................. 61

7.2 Speech Primer with Musical Model ................................................................. 64

7.3 Successive Primers Process for Etude III. ......................................................... 68

7.4 Etude III. Annotated Score ........................................................................... 69

7.5 Etude V. Injection Tracing ............................................................................. 73

7.6 Etude IV. Angry Speech Spectra Primer ......................................................... 76

7.7 Etude IV. Injection Comparison .................................................................... 78

7.8 Etude VII. Theme ......................................................................................... 80

7.9 Etude VII. sequence three ............................................................................ 80

7.10 Etude VII. sequence five ............................................................................. 81

7.11 Etude VII. sequence eight ........................................................................... 81

7.12 Etude VII. ending ....................................................................................... 82
Abstract

This research addresses the development of machine learning techniques used to create musical scores and performances that are inspired by the intersection of speech and music. Machine learning models are created from MIDI files that are transcribed from datasets of musical audio recordings and human speech audio recordings. Through the creation of succinct models, model based cross synthesis is possible. Models trained on musical MIDI data are asked to replicate MIDI data that approximate human speech. Alternatively, models that have been trained on MIDI data that approximate speech are asked to replicate musical MIDI data. The product of these developed techniques is a collection of piano music, *Seven Piano Etudes Speaks the Moody Machine*. These etudes are intended to be performed on one Yamaha Disklavier piano with two performers, one human pianist and one machine player piano.
Chapter 1
Introduction

Human speech is an intrinsically musical phenomenon. The musicality of speech has aroused the interest of such scholars as Plato[26], Charles Darwin[14], and modern researchers like Aniruddh Patel[33][34]. As perceived by the human ear, spoken-word contains extended meaning and emotional expression. The expressive qualities of music and speech overlap. Rhythmic, melodic, intervalic and spectral qualities of speech contain a power to convey human expression strikingly similar to musical expression.

Music and Speech are tools used for communication. Music is a means of communicating expression that is subjective in interpretation. Language exists as a tool for conveying information. When language becomes spoken-word there is meaning beyond the literal interpretation of words. This non-literal aural expression exists in music and speech.

Musical qualities in speech and speech-like qualities found in music are valuable sources of inspiration and creation. As a result, techniques to create musical scores and performances that are inspired by the intersection of speech and music are fruitful means of composition. In the history of electroacoustic music speech has played an important role in composition. These roots date back to such works as Luciano Berio’s Thema (Omaggio a Joyce) and Karlheinz Stockhausen’s Gesang der Jünglinge.

However, in recent years the advancement of machine learning as a tool for creation has provided a powerful means for exploration of this unique relationship. Machine learning models are trained using recurrent neural networks to generate musical material inspired by speech. The corpora of these models are composed of symbolic music data in the form of MIDI files. These models fall into two overarching categories: those trained on corpora containing traditional MIDI files used to represent musical information and models trained on novel MIDI files that have been created to imitate the rhythmic, melodic and spectral characteristics of speech. A wave to MIDI file conversion tool creates MIDI encodings from
audio files of speech. Several moods to include amusement, anger, and neutral speech are trained into corresponding models, each with unique speech characteristics. The musical models are based on curated corpora to include the works of various modern composers and improvisers as well as the author’s own improvisational style.

Through the creation of models based on speech and those based on music, cross synthesis is possible. These trained models can then be assigned MIDI primers which inspire models to generate MIDI material that resembles the primer file as seen through the lens of each model’s training examples. Speech primers fed into models trained on music and music primers fed into models trained on speech, allow for the examination of sonic possibilities that exist in the middle ground. Additionally, interesting MIDI files can be generated by interpolating between these various models within a single piece of music. These techniques, as represented in figure 1.1, allow for the generation of sonic landscapes that hybridize music and speech in the form of musical scores for compelling human and machine performances alike.

With the exploration of the aforementioned methods, a body of curated scores are created and presented as a portfolio of piano works entitled, *Seven Piano Etudes Speaks the Moody Machine*. These works are intended to be preformed in a unique piano duo that features one human pianist and one self playing Yamaha Disklavier Piano performer.
Figure 1.1: MIDI models trained on music and speech
Chapter 2
Relationships between Speech and Music

2.1 Origins of Music in Relation to Speech

Every theory regarding music’s origins deals with the effects, purpose, and value of music in its most general relationship to humanity.[26] It is a subject that has peaked the interest of many of history’s greatest thinkers by the inherent relationship between its origins and current manifestations. The origins of speech are intrinsically tied to those of music. Each theory regarding music’s origins can be grouped into three camps of thought: those which believe that human speech developed before music, those which believe that music came before speech and those that believe that speech and music emerged simultaneously. Additionally, these theories are best understood as arguing one of three main ideas: music is a product of spontaneous emotional expression, music is linked to human sexual instincts, or that music is the product of innate rhythmic tendencies.

Plato’s Cratylus is the starting point for most arguments regarding music’s origin. Plato saw music as an extension of ethics and regarded its purpose as an imitation of defence, aggression, and persuasion. All three of these are summed up by Plato as means of achieving social harmony. This type of thought is not exclusive to Plato in the ancient world. Confucius believed music was a natural force in the universe and that it had the power to unite people. A similar idea of unification is presented in the book of Job, “when the foundations of the earth were laid, when the morning stars sang together, and when all the sons of God shouted for joy”.

Giambattista Vico, who was no doubt influenced by Plato, believed that humans first words were imitative of the natural environment and emotionally expressive in a spontaneous manner. According to Vico this expressive quality resulted in early language being sung and not spoken.

The French philosopher Étienne Bonnot de Condillac also believed that human speech
was the result of spontaneous expression. However, he makes one distinction by giving further reasoning in support of early language being more akin to music. His reasoning was a biological argument that early humans had undeveloped vocal apparati that could not resemble modern speech. For this reason he believed that early speech was less sophisticated than modern speech and more musical in resemblance.

In his Essay on the Origin of Language, Jean Jacques Rousseau argued that the early human need for language resided in morality and passion. His ideas stemmed from the idea that society itself is the result of humans concerning themselves with the well being of other humans. Like Condillac, Rousseau believed that early speech existed as a proto-music. However, he did not believe that these sounds were in any way an imitation of the environment and that instead these sounds were innate in humans and expressed sadness, joy and all emotions needed to show pity or rejoice in the experiences of other humans.

Charles Darwin is the chief proponent among thinkers who favored the idea that the origins of music are related to sexual instincts. His ideas about early speech allow for early human use of gestures, imitation of nature and “man’s own instinctive cries”. He also asserts that early peoples would use their voices primarily for singing and that this singing served some purpose in courtship for procreation. He separates these courtship vocalizations into two realms: those seeking to attract mates and those used to display dominance among rival males. His theories are based on the behavior of other animals that “produce musical notes or mere rhythmical sounds” to attract mates.

Another scholar, Herbert Spencer, believed that Darwin had it backwards. In Spencer’s argument music was an imitation of emotional speech. He believed that music was exaggerated and idealized passion in natural language. Interestingly, he also believed that music, dance and poetry were originally all one thing.

The starting point for the idea that music is the product of innate rhythmic tendencies begins with Richard Wallaschek. While Rousseau maintained that rhythm grew from the same periodic sources that produced melody, Wallaschek argued that an instinct for rhythm
was an innate feature of any organism. Furthermore, he postulated that rhythmic insentives created a desire in early humans to develop melody and harmony as musical features that initially served to enhance the rhythmic experience. He makes a good point by explaining that melodic intervals have no value without their ordering in time.

Amongst all of these theories the common denominator is that human connectivity is the reason for speech and music. Music is either the genesis for speech, speech is the genesis of music, or speech and music emerged simultaneously in human history. Either Way, the link between the two is undeniable and interesting. Music is by any description a link to our primal ancestry. To better compare speech and music it is beneficial to consider speech in terms of the musical components of rhythm, melody and intervalic relationships.[26]

2.2 Rhythm in Speech

The Rhythmic qualities of language and music have been a focus of scholarship for linguists and musicologists for a long time. There has been dispute among scholars as to the appropriateness of using the word "rhythm" in relation to speech. However, speech is definitely dependent upon and effected by time. The idea of rhythm must be defined if it is to be considered as a feature of speech. Rhythm is often thought of in two ways: with an emphasis on regularity or with an emphasis on duration relationships. The regularity perception of rhythm constitutes the presence of a regular pulse. This is the same devise that allows for hand clapping or foot tapping to music in a traditional context. While this is often found in speech in an approximate way, considering duration relationships is a more productive method of conceiving of rhythm in speech.

One method of examining duration relationships between speech and music was proposed by the cognitive psychologist Aniruddh Patel. Patel’s method for comparing musical rhythm and speech rhythm is to use a linguistic duration measurement tool to analyse musical rhythm.[33][34]

An empirical comparison between rhythm in speech and music is dependent upon the correlation between the features of these two domains. In an effort to reconcile the
differences in perceived rhythm in speech versus music Patel employed a linguistic analysis tool known as the “normalized pairwise variability index”, or nPVI. nPVI reveals the amount of duration variability of stressed and non-stressed vowels in speech.

In former studies nPVI has been used to compare intrinsic rhythmic differences between French and English. These two languages were chosen because they have long been thought to differ greatly in rhythmic tendencies. The general belief is that in spoken French, stressed vowels receive relatively the same duration as non-stressed vowels. In English on the other hand, stressed vowels are commonly given a longer duration than non-stressed vowels. Therefore, the nPVI is much higher in spoken English than in French. One study by Franck Ramus, Marina Nespor and Jacques Mehler[35] showed that English examples observed ranked an average nPVI of 66.99, while French showed a nPVI of just 49.27.[33]

As a mean of comparison Patel applied the linguistic tool of nPVI to melodic musical themes. In each theme the first note of these themes was assigned the duration of 1 and all following durations were expressed as fractions or multiples of the first duration.

The musical corpus of French and English examples was composed of melodic themes by French and English composers of the “musical nationalism” movement from near the turn of the 20th century. The reasoning for this was that these examples might better reflect musical material that is idiomatic to the regions. The examples from these corpus were converted to nPVI directly from notated music.

The results of Patel’s study indicated that the English corpus was assigned a mean nPVI of 46.91 and the French corpus was ranked with a mean nPVI of 40.90. As a result the comparison between these two languages held true. English music and speech both had higher rhythmic duration variability than French music and speech. This scholarship is evidence that regional linguistic traits translate into regional musical traits. Additionally, this is demonstrates that rhythm in music and speech are linked and comparable[32]
2.3 Pitch in Speech

There is evidence of a biological link between the pitch material that comprises music and speech. Human hearing allows for the distinction of around 240 pitches within the mid-range of audible sound.[17] In spite of this, collections of pitches such as scales which are used to create music contain on average between five and seven tones if octave equivalency is considered. This phenomenon is mysterious. However, research shows that the genesis of our small pitch collections may be a result of the evolutionary advantage afforded to humans though perception of the spectral content of human speech.

Research regarding the origins of musical pitch material is not a new pursuit. Von Hoerner’s work has shown a reasonable explanation for the 12 tone chromatic division of the octave. It’s necessity is linked to tonality. According to his findings, harmonic polyphony requires that higher overtones created by lower fundamental pitches bear a close relationship in frequency to the fundamental frequencies of higher notes from the scale. Although splitting the octave into 12 divisions is not the only solution to this, it is the most feasible for the human auditory system. Additionally, it is more practical for technical considerations in performing instrumental music. According to Hoerner, this is the reason that musicians from the seventeenth century arrived at the tempered chromatic scale we know today. However, this explanation accounts for our chromatic scale and not even smaller pitch collections.

There have been attempts to explain smaller pitch sets. Bernstein, perhaps most famously, explained that the intervals comprising the major pentatonic scale are the same as those of the first nine members of the harmonic series.[41]

Similarly, a quantitative study conducted by Gill and Purves[17] indicates that there is considerable evidence that the scales used to make music throughout many cultures and the history of music are linked by their similarity to the harmonic series. Furthermore, their study shows that musical pitch collections preferred by humans are of a particular relationships to the harmonic series. Because these findings are common across cultures
and time, the argument presented by Gill and Purves hinges upon universality through biological utility. Various animal species produce periodic sounds that are a type of communication and most importantly focus on successful reproductive tendencies. It is logical that human speech would have evolved for the same purposes. Additionally, the human ear canal and basilar membrane are perfectly suited to perceive the spectral content that can be produced by the vocal tract. Therefore it is likely that the co-evolution of speech and hearing are direct descendants of the spectral content of musical pitch collections and scales. Additionally, the fact that music mimics speech is a possible explanation for why music is enjoyable for humans.[38][17]

The melodic nature of speech has also been considered from a psychological perspective. Diana Deutsch’s discovery of what she calls ”The speech-to-song illusion” relates to melody in every day speech.[10] The speech-to-song illusion suggests that if spoken words or phrases are repeated or looped they will be perceived by the listener as melodic musical phrases. An explanation for this phenomenon has not been proven. Even as a mysterious observation the connection between melody in music and speech is strong.

2.4 Intervalic Relationships in Speech

The presence of pitch simultaneity in speech is less obvious than rhythm or melody. However, speech like any other periodic sound contains harmonic relationships in its spectral content. More importantly, speech maintains a unique harmonic property through ratios of its first two vowel formants. In the fields of phonetics and speech science, formants are the result of the natural resonance of the vocal tract. The vocal tract has two important resonating chambers that produce formant one (F1) and formant two (F2). F1 is produced in the space between the top of the larynx and the tongue. F2 is created between the tongue and lips. Both of these resonating chambers are particularly important for vowel phones. Without these two features, speech is indistinguishable. F1 ranges in frequency between 200 and 1,000 Hz while the F2 resonates between 800 and 3,000 Hz.[38]

A study conducted by Dale Purves[38] suggests that musical intervals that form the 12
tone chromatic scale resemble harmonic ratios between the first two formants used in vowel pronunciation. Purves maintains that the human preference for these intervals in music are the result exposure to theses intervals in normal vocal communication. To investigate this phenomenon a study was conducted by analysing speech formants of native English and native Mandarin speakers. Recording and analysis of single word and monologue excerpts were considered. For English, participants spoke eight words with different vowel sounds between the letters “b” and “d”. Mandarin participants spoke six words representative of the vowel sounds in Mandarin (ba, ge, bo, bi, du, and ju ). Participants from both language groups read allowed five fifty word monologues in a neutral tone. The results showed intervallic relationships representing all interval combinations produced by the chromatic scale. Sixty-eight percent of the collected intervalic combinations are intervals percent in the 12 tone cromatic scale. Additionally, 70 percent of these are found in the pentatonic scale and 80 percent represent the diatonic scale. The conclusions drawn by this study suggest that the global preference for cromatic, diatonic and pentatonic scales is the result of exposure to these formant ratios in speech.

2.5 The Musical Tendencies of Emotional States of Speech

Human speech expresses the literal meaning of words as messages. Additionally, spoken word communicates emotional content that further informs these messages. Music without words on the other hand does not normally include literal messages. Instead, it’s message is much more subjective. Often musical messages are interpreted by listeners as expressing certain moods or emotions.

The emotional impact of music is the product of many factors. A composer or performer might express particular moods by employing musical devices such as tempo, rhythm, dynamics, density, harmony and melody. Although all of these factors are subjective contributors to emotional expression, musical analysis tools have been developed to understand the elements that create emotional expression in European classical music. It is an interesting idea that the emotional impact of speech might be measurable though means which
are usually used to measure and analyze music and the emotional moods created by music.

Major and minor tonality is one method of musical analysis which is often tied to mood. Music in major keys tend to be interpreted by listeners as excited, happy and bright. On the other hand music in minor keys tend to be perceived as subdued, sad, or dark. The reason for this is not clear.

One study by Daniel Bowling[6] presents a possible explanation by comparing major and minor music to excited and subdued speech. With their findings, they theorize that certain interval relationships in music are tied to certain emotional states because those same intervals express the same emotions in spoken language.

To demonstrate this researchers compared the spectral content of audio recording databases of western classical music in major and minor keys to the spectral content extracted from audio recordings of speech performed in excited and subdued emotional states.

The speech database was created by recording the voices of actors. These actors were recorded speaking single words and sentences in either excited or subdued states of speech. The spectral content of these recordings was extracted by an auto correction algorithm which allowed for the examination of speech formants. The difference between the F1 and F2 was calculated as a parallel to the harmonic intervals found in music.

To begin, the databases of music in major keys was compared to those in minor keys in a effort to find distinguishing characteristics that could also be applied as a way to differentiate excited and subdued speech. This was achieved by calculating the frequency of each interval’s occurrence as a percentage of the total number of intervals counted.

The results of these calculations showed that the determining factor in the classification of music as major or minor was the number of major or minor 3rds that occurred in a given piece of music. In the music database major thirds made up between 16 percent and 18 percent of the total intervals in major pieces of music. Additionally, minor 3rds made up less than 1 percent of the total intervals in pieces in major keys. These results were reversed
when examining the interval content of music in minor keys where minor 3rd made up 15 percent of all intervals and major 3rds occurred less than 1 percent. Other intervals were examined but 3rds proved to be an effective measurement.

When observing the intervalic ratios created by the first two formants, intervals were then calculated as F2/F1 and were distinguished as chromatic if they were within 1 percent of just intonation. Through this calculation, excited speech showed many major interval ratios and relatively few minor interval ratios. Subdued speech on the other hand was composed of fewer major interval ratios and more minor intervals. Formant ratios corresponding to major seconds, thirds, sixths, and sevenths made up 36 percent of all intervals in excited speech. Intervals such as minor seconds, thirds, sixths, and sevenths were not found. In subdued speech just 20 percent of the intervals were major seconds, thirds, sixths, and sevenths. Oppositely, 10 percent of the ratios corresponded to minor seconds, thirds, sixths, and sevenths.

The conclusion drawn by this study was that the spectral content of excited speech more closely resembles music in major keys and the spectral content of subdued speech better resembles music in minor keys. These findings point to a possible explanation for the association of music in major keys as more excited and music in minor keys as more subdued. Furthermore, this conclusion highlights the musicality of speech.[6][22]
Chapter 3
Historical Context

3.1 Speech in Musical Composition

The compositional output of music related to human speech in electroacoustic music is vast and remarkable. By the turn of the 20th century the groundwork for what would become the use of speech in recorded music was marinating. In Italy, the futurist movement was alive with ideas that would shape this practice. Marinetti’s *Parole in Liberta*\[28\] abandoned the actual meanings of words in favor of text arranged for the sake of their sounds. However, until recently text and speech were an oral tradition and one that could only be recorded with the pen or in print. The advent of sound recording and its eventual accessibility would change things and blur lines between sound art, the words themselves. Works such as Luciano Berio’s *Thema (Omaggio a Joyce)* in 1958 manipulated the reading of text from James Joyce’s *Ulysses* by means of overdubbing, filtering and speed manipulation. Charles Dodge’s work on the first of his Speech Songs were early examples of synthesizing speech. By the summer of 1972, Dodge had the realization that making “sung” vocal computer music was not as feasible or as interesting as making speech music.\[7\] Other works like Alvin Lucier’s *I am Sitting in a Room* used human speech to explore the resonant frequencies of space through recorded speech.

While the aforementioned speech works are important in the history of electroacoustic music, they do not directly relate to the extraction of musical characteristics from speech. A survey of works that showcase the rhythmic, melodic and spectral aspects of speech is necessary in the interest of observing a historical context.\[43\] Among composers who deal with the rhythmic qualities of speech, Steve Reich is a great example. Specifically, his early tape-based works such as *It’s Gonna Rain* (1965) and *Come Out* (1966) deal directly with spoken-word in new rhythmic ways. In *It’s Gonna Rain*, the audio source material for the work is a recording of Pentecostal preacher, Brother Walter. The recording itself
is very compelling. Reich explained the recording well, “Sometimes when people speak, they almost sing. Tape loops are little bits of tape that are spliced together so that they just go around and around and around and repeat themselves. And when you take a bit of speech like ‘It’s gonna rain,’ the way he says it, you really begin to hear the music of what he’s saying and what he says increasingly blended together so it’s hard to separate them.”[39] With these pieces Reich examined phasing in which speed variations of different tape loops caused a flanging effect and gradual rhythmic separation.[40] An example of the melodic aspect of human speech can be found in Reich’s 1988 masterpiece, *Different Trains*. Unlike his early tape pieces this work is for tape and string quartet. Reich uses interview recording of just a few words to create the melodic and rhythmic material of the piece that is performed by the ensemble.[15]

Another example of melodic vocal emphasis is Karlheinz Stockhausen’s *Gesang der Jünglinge* (1955). In this piece Stockhausen commingled recorded voices with pure electronic sounds. To do this he separated the sung verses of a child into their elementary phonetic components. By studying phonetics and observing spectral elements, Stockhausen was able to understand the overtone structure of sung vowels versus consonants. While vowels resemble pure tones, consonants resemble noise. With this in mind Stockhausen created sine waves that imitated vowel sounds. Conversely, he used generated noise to produce consonant sounds.[30]

The music of Paul Lansky is perhaps the most well-known computer based music that deals with human speech. These pieces are great examples of speech used to create harmonic material. Instead of drawing harmonic material from speech he used synthesis to re-pitch vocal content into harmonic instances that suited his compositions. Lansky’s plunge into text-based music began with a piece called *Six Fantasies on a Poem by Thomas Campion*. In *Six Fantasies* he combined processed voice recordings of his wife Hannah MacKay reading the Poem and synthesized her speech into textures of harmony. Lansky’s *Chatter* pieces dealt specifically with recorded voices in which he broke apart words into their distilled
syllables and reconstructed them in harmonic ways which are often at least based in the tradition of western tonality.[37]

There are a few composers who have dealt with all three examined traits of speech: rhythm, melody and harmony. Trevor Wishart explored the sonic phenomenon of human speech in a vast number of ways. In the 1990s, Wishart began writing for the human voice with his famous Vox Cycles. Later speech works include Tongues of Fire and the Voiceprints Cycle in 2000. In general Wishart’s primary interest in the human voice was what he referred to as sound transformations, whereby recorded sounds are imitated, synthesized and interpolated by the voice. In accomplishing these sound transformations Wishart organized human speech into what he called “sound objects”. The term “sound objects” was conceived by Abraham Moles as a means of referencing a primary source audio recording which could be used as tool for composition.[19] For Wishart, these sound objects are small sets of sound syllables that are common to all languages. Wishart's description of these sound objects includes a detailed understanding of the melodic, rhythmic, harmonic and timbral qualities of speech.[45]

More modern uses of the rhythmic, melodic and harmonic aspects of speech in music can be found in the music of Peter Ablinger. Peter Ablinger describes his process as “phonorealism”, which takes its name from photo-realistic painting. In essence Ablinger developed a speaking piano. His creation uses 88 robotic electromechanical fingers that can perform mappings of voice recordings. To create these mappings, he utilized spectral analysis of recorded speech to extract key features of speech. The result is a mechanical performer that recreates roughly recognizable speech through the mechanical performance of an acoustic piano.[3][36]

Jason Moran is a jazz pianist and band leader who began his solo career in 1999 with the album Soundtrack to Human Emotion. He has won many awards for his playing and compositional skills which showcase his unique mixture of stride piano, avant-garde jazz, classical music, and hip-hop. Moran uses the characteristics of human speech as a vehicle
for group improvisation. One example is the piece *Ringing My Phone* in which Moran’s piano trio performs live with a recording of a woman speaking in Turkish. In performance the trio matches the rhythmic, melodic and harmonic content of the recording. As an ensemble the trio memorized as Moran put’s it “the speech and the breath.”[46]

3.2 Algorithmic Symbolic Musical Composition

Algorithmic composition could be defined as “the process of using some formal process to make music with minimal human intervention”[2]. This idea is an old one and includes notable innovations from the likes of ancient Greek musical thought, Mozart, and John Cage, to name a few. However, the advent of the computer allowed for an explosion of algorithmic composition that can be broken down into three categories: stochastic, rule based, and artificial intelligence. In these examples, the computer had no part in generating the actual sound performance. Instead, algorithms were used by the composer to realize compositions as scores to be performed by humans.[29]

The first computer generated composition was achieved by Lejaren Hiller and Leonard Isaacson in the mid 1950s. The result of their work was a string quartet, *The Illiac Suite*. This composition is especially interesting because the material of the piece was composed by the computer and then translated into a music score in standard notation. The process engaged by Hiller and Isaacson to create this piece was rule based and consisted of three steps which Hiller and Isaacson called subroutines: the generation of raw material, the modification of material and the curation of results. This three-step process was further developed by Hiller and Robert Baker as a library of subroutines in their program MUSICOMP in the early 1960s. Because these subroutines were customizable it allowed for much more flexibility and the creation of stylistic options for the composer.

A great example of stochastic composition can be seen in the work of Iannis Xenakis. His work took advantage of the computer’s high computational potential as a means to calculate probability theories. Xenakis’ program allowed for the realization of a score from lists of note densities and probabilistic weights which he set. These decisions were then
combined with random number generators resulting in compositions such as *Atrees* and *Morsima-Amorsima* in the early 1960s.

One final algorithmic compositional technique is that of artificial intelligence. It is similar to rule-based processes in that these systems follow some predefined grammar. However, they are also capable of learning and creating new grammar which may not be known by the composer. The most famous example of this is David Cope and his creation Experiments in Musical Intelligence (EMI). In this system the program was fed scores from a specific composer’s catalog. Based on these scores, EMI was able to create its own rules and learn stylistic features that resulted in very convincing new works in the styles of composers such as Mozart and Bach as well as many others.[29][9]

The most significant innovations in artificial intelligence in the last five years involves an emerging field known as “deep learning” which functions with and benefits from extremely large data sets.[4] Through many examples, machine learning is enhanced in a manner that allows neural networks and other architecture to intricately define structure. The impact of deep learning has created much potential innovation in the world of art, to include musical composition.

Google’s Magenta is currently at the forefront of deep learning musical composition. Magenta is explored in depth in chapter four.
Chapter 4
Magenta: Generating and Training With Recurrent Neural Networks

4.1 Magenta

Magenta is an open source Python library designed to investigate the possibilities of machine learning in artistic creation. It was designed by the Google Brain team and uses TensorFlow, another Google library, to ease the process of building datasets, training models, and creating predictions. The Magenta library includes tools to sort music and image data to train machine learning models and generate new art.[12]

Magenta’s developers propose three goals. Firstly, it’s goal is to advance machine intelligence for music and art generation. In addition, Magenta hopes to give attention and access to many users hoping to explore the possibilities of machine learning art. Finally, Magenta hopes to build a community of artists, programmers and researchers.

The evaluation of generative models is difficult. This is particularly true when art is the result of these generations. In more normative machine learning contexts the average log-likelihood is used to calculate the amount of deviation of generation from training data. In art this is problematic for two reasons. If deviation is none or little the generation might be an exact near exact replica of the training data. If the dataset is created on musical scores of J.S. Bach and the resulting generation is the same content as the first prelude from the Well Tempered Clavier, nothing artistic has been accomplished. The interest in machine learning music generation occurs when a model yields generations that are similar to the training data but are unique in some way. Additionally, generations might resemble training data in a certain way and fail to perceive features that a human listener might find especially musical. Human ears can evaluate the virtue of generated art in a profound way. Still, the idea of a musical turing test is an interesting phenomenon.
4.1.1 Magenta’s Contents

There are many ways of representing sound data. Magenta’s content takes advantage of these various vantage points such as symbolic data, spectrogram data, and raw audio to name a few. Symbolic data is the prominent medium for music generation with Magenta. Symbolic musical data is commonly represented in three formats: MIDI, Music XML and ABC Notation. Magenta can use all three. Among these three MIDI is an ideal candidate for creation since it is extensively supported and can easily be used to create visual scores for machine and human performance.

Among the Magenta models dealing with symbolic data, there are five overarching categories: rhythmic generation, melodic generation, polyphonic generation, interpolation, and transformation. Drums RNN uses language modeling with LSTMs to create drum beats. Melody RNN uses attention masks to create monophonic material that is very good at dealing with longer segments of symbolic data. Polyphony RNN is very similar to Melody RNN except it has certain features that allow for polyphony. Another polyphonic model called Performance RNN attempts to humanize the symbolic data by implementing expressive timing and dynamics. Music VAE allows for interpolation between existing monophonic sequences. GrooVAE is a model that interpolates in the same way as Music VAE but adds the expressive timing and velocities as referenced in Performance RNN. Among these Polyphony RNN is best suited to cross synthesize machine learning models of speech and music since polyphony is desired and training is more easily accessible that Performance RNN. Performance RNN is particularly interesting because it generates MIDI with expressive timing and dynamics. However, the dataset used for the pre-trained model is the Yamaha e-Piano Competition dataset. This data set contains over 1400 MIDI performances by skilled pianists. While this allows for interesting MIDI representations of piano performance, there is no known equivalent means of creating such a dynamic speech dataset. In an effort to better understand music and speech cross syntheses, recurrent neural networks and Magenta’s Poliphony RNN is explained and employed.
4.2 Recurrent Neural Networks

Neural networks (NNs) can be composed of various architectures. For the purpose of music production recurrent neural networks (RNNs) are ideal for two reasons. RNNs are great at operating with sequences in regard to inputs and outputs. Additionally, they are more well suited than other NNs at contextualizing past events with current predictions. In other words, RNNs are good at keeping track of time and are therefore well suited for music since sound is time dependent for aural perception.

4.2.1 Sequencing Vectors

RNNs operate with sequences of vectors. These sequences can be realized in many ways. Specifically, RNNs are beneficial because their input and output lengths are flexible in size. One to one sequences work with fixed input and output vectors. An example of this is image classification. One to many sequence outputs take one input such as an image with the output being text that represents the image content. Many to one sequences outputs take an input sequence and output a single result such as a classification. Many to many sequences outputs might accomplish something like language translation where one sentence in one language results in a sentence in another language. [12]

The most common way to explain the architecture of a RNN is using a diagram. In the figure 4.1 the three layers of a RNN are represented containing an input layer, a hidden layer and an output layer that feeds into itself. The bottom row indicates input vectors, the middle row represents hidden layers and the top row shows the output layers.
Figure 4.1: RNN Vectors
In a Convolutional Neural Network (CNN) information is fed in one direction, from input to output. This is what is known as a feed-forward neural network. Each step of a CNN’s output is not affected by previous events. RNNs on the other hand manage previous events in each step’s output. RNN architecture accounts for the states of previous outputs in each step with what is commonly known as the hidden layer. In each step and input vector, x, yields an output vector, y. Additionally, the hidden vector, h, is updated by updating the loss. For each sequence in a step the RNN generates a confidence level as to what the next sequence might be composed of based on both the input and the hidden layer which remembers the outputs of previous steps. The hidden state is updated with each step based on every former output. For this reason each additional step contains an updated hidden state that with the current input is more likely to predict the next step based on previous outputs. The continuous process of updating hidden states based on previous output is what’s known as back propagation. For example, if the input data is the first four notes of a C major scale (C, D, E, F), a four step vector is encoded. This four note vector must be broken down into four individual steps in chronological order. The first step being C can be encoded as [1, 0, 0, 0] which corresponds to x(t -1) in figure(n). The next step can be represented as [0, 1, 0, 0] which corresponds to x(t) in figure(n). These two steps would be followed by t(x + 1) encoded as [0, 0, 1, 0]. In the first step the RNN might give the following probabilities of prediction: 0.5 for C, 1.8 for D, -2.5 for E, and 3.1 for G. Since the training data provides that the next step is D, the probabilities are adjusted accordingly by increasing the likelihood of D and decreasing the likelihood of other predictions.

4.2.2 LSTMs

As RNNs backpropagate and adjust the hidden state gradients of the hidden state become smaller and smaller. When sequences are long enough these hidden state gradients are multiplied many times by very small numbers and eventually disappear. To deal with this problem Long-Short Term Memory (LSTM) cells are employed within the RNN. On
the most basic level LTSMs take previous layer output, concatenate it with current input, and use a function such as tanh or sigmoid to create the layers output and the next layers input through a series of gates. LTSMs have three gates, to protect and control the cell state: the forget, input and output gates.[12]

Figure 4.2: LSTM Cell

4.3 Magenta’s Polyphony RNN Generation Algorithm

4.3.1 Encoding MIDI Data

Every Magenta Model used to create scores uses a class named NoteSequences. Through the use of this class MIDI data can be encoded and used for model training data and subsequent generation. NoteSequences uses a language neutral method for serializing structural data called, Protocol Buffers or Protobuf for short. The information handled by NoteSequences is MIDI message information to include: time signatures, key signatures, tempi, and note lists. In monophonic Magenta Structures such as Melody RNN, notes are organized as lists of pitches, which refers to MIDI note numbers that are used with simple start_time and end_time information in reference to each note.

4.3.2 Polyphonic Encoding

Among all RNN architectures created with Magenta, Polyphony RNN is ideal for cross-synthesizing MIDI files created in the likeness of music and those created in the likeness of speech. In fact it’s design was created in a similar manner to traditional language
modeling with the use of an LSTM. Polyphony RNN is capable of generating multiple simultaneous notes. It’s inner workings are similar to another RNN designed for symbolic music generation called BachBot.[27]

Much like BachBot, Polyphony RNN works as one string of notes. There are a total of five different event classes to encode data: START, STEP_END, NEW_NOTE, CONTINUE NOTE and END symbols. Pitch is represented as MIDI note values and notes are sorted by pitch in descending order within each step of generation.

The following is an example encoding notes from a G Major chord with a duration of one quarter note since each step represents a sixteenth note. With Polyphony RNN note endings are not specified in encoding. Here each note is continued with the next step. If CONTINUE_NOTE did not a precede the note 67, the encoding would not create it in that step:

```
START
NEW_NOTE, 74  
NEW_NOTE, 71  
NEW_NOTE, 67  
STEP_END
CONTINUE NOTE, 74
CONTINUE NOTE, 71
CONTINUE NOTE, 67 
STEP_END
CONTINUE NOTE, 74
CONTINUE NOTE, 71
CONTINUE NOTE, 67 
STEP_END
CONTINUE NOTE, 74
```
4.3.3 Arguments and parameters

The generation algorithm works by iterative predicting each generation step in a sequence based on what the model has learned during training. Because we are interested in generating symbolic musical information this means that each note or rest is predicted sequentially until an entire score of a desired length is created. The scalable features that determine how the model will generate a sequence are: beam size, branch factor, and the number of steps per interaction. The function of each of these options is best understood through an example.[12]

Consider a generation with the following command line parameters.

```
polyphony_rnn_generate --bundle_file=NewMelodies.mag --
output_dir=output --temperature 1.0 --beam_size 1 --branch_factor 2 --
steps_per_iteration 1 --num_steps 32
```

In this example `polyphony_rnn_generate` instructs Magenta to use the melody_rnn architecture for this generation. `Bundle_file=NewMelodies.mag` points to the specific training data referenced for generation. The `output_dir=output` tells the model where to save the generated sequence once the process is complete. The `num_steps 32` instructs the model to loop the generation process 32 times, resulting in 32 steps. Since each step is by default a sixteenth note, the number of steps per iteration is 1 and the model defaults to 4/4 time, the generation will be two measures in length. With a `beam_size` of 1 and a `branch_factor` of 2 the model chooses the best candidate from two generated sequence steps.

Through Magenta’s installation process an Anaconda environment is created which allows for console entry points into Python scripts contained in Magenta’s source code. These
can be executed as command line utilities. Among the most fundamental command-line utilities create_dataset, generate, and train. These commands always contain a prefix that specifies the Magenta model used. For example drums_rnn_generate is the command to generate a new MIDI file from the drums_rnn pre-trained model. More specifically drums_rnn refers to a bundle file, drums_rnn.mag. These bundle files contain model checkpoints and metadata from this pre-trained model. The following is a list of flags that can be used to set the parameters of any of Magenta’s RNN architectures:

- **–num_outputs.** Specifies how many generations to be created per execution. These generations are MIDI files.
- **–num_steps.** Specifies the length of the generated MIDI file in steps. Each step equals one 16th note. Because of this a measure of 4/4 time will have 16 steps. If a primer file is used these steps are in addition to the number of steps contained in the primer file.
- **–qpm.** Specifies the number of quarter notes per minute (QPM). In Magenta QPM is a way of controlling tempo. If a primer file is used the QPM of the primer file will be used and this flag will be ignored.
- **–primer_mid.** Specifies the path to a primer MIDI file. The number of steps of the generation must be longer than the number of steps in the primer file.
- **–temperature.** Specifies the randomness of a generation. A temperature of 0.1 uses softmax possibilities, a number higher than 0.1 makes the generation progressively more random and a number lower than 0.1 makes the generation less random.
- **–beam_size.** Specifies the beam size to use for mean search.
- **–branch_factor.** Specifies the branch factor to use for bean search.
- **–steps per iteration.** Specifies the number of steps to talk per beam search iteration (default 1)
4.3.4 Primers

In Magenta primer files are an interesting way of inspiring model generation. As a parameter, a priming MIDI file can be set before generation. When set the model will attempt to create something similar to the primer with the only tools it has, its training data. This is the primary way of creating model based cross-synthesis. For example, if a model has been trained on MIDI that represents human speech and a primer file of a musical example is set, the speech model will attempt to create something similar to the musical example through the lens of its training data, speech. Conversely, a musical model primed with a speech MIDI file will result in a generation attempting to imitate speech through a musical language.

Additionally, two other parameters can be set to control how the model uses the primer MIDI file. Condition on primer and inject primer during generation are both boolean statements. If condition_on_primer is set to true, the primer sequence will be used by the RNN for generation. Inject_primer_during_generation on the other hand tells the network to begin the generation with the MIDI primer in the actual generation output. If a primer is set for a generation and condition_on_primer and inject_primer_during_generation are both set to false, the primer file will appear in the generated MIDI file before any newly generated material.[12]

4.4 Training

Training machine learning models is comprised of a few steps. First datasets of MIDI files must be created that contain features that the model can learn from. Once a dataset is collected, MIDI files must be converted into notesequences. This is done in a command line action, convert_dir_to_notesequences. The result of this command is the creation of the file, notesequences.tfrecord. Next the sequence examples can be launched by another command line action that points to the newly created notesequences.tfrecord document. This action creates a sequence_example folder that contains a training set and an evaluation set.
Finally, the actual training can begin. The following command line prompt tells Magenta to use the Polyphony RNN architecture, sets the output directory with run_dir, points to sequence examples, asks that the network for training be a three layer with 128 units per layer and a overall batch size of 64. In addition this command is calling for the network to train for 30,000 steps.[12]

```
polyphony_rnn_train \
  --run_dir=tmp/polyphony_rnn/logdir/run1 \
  --sequence_example_file=tmp/polyphony_rnn/sequence_examples/ 
    training_poly_tracks.tfrecord \
  --hparams="batch_size=64,rnn_layer_sizes=[128,128,128]" \
  --num_training_steps=30000
```

4.4.1 Bundle Files

Bundle files are used to package a tensorflow checkpoint, metagraph, and some metadata about the model into a single file in a file that can be referenced for generation. In tensorflow checkpoints contain information regarding the models training. This facilitates recalling and sharing a specific point in the model’s training. Bundle files can easily be created by users with a command line execution after training.[12]
Models Created Representing Traditional Musical Scores

Machine learning models based on speech and those based on music make cross synthesis of speech and music possible. The musical models are based on MIDI transcriptions of recorded piano music. Collections of these transcriptions form corpora that when trained into machine learning models, emulate musical styles of select composers and improvisers. This compositional process is comprised of four models of different piano styles based on these musical artists or works: $D(\mu)\theta$ by Mara Gibson, the compositions and improvisations of Thelonious Monk, the piano playing of James Carroll Booker III, and piano improvisations by the author, William A Thompson IV.

The evaluation of this technique is difficult. In the identification of how well this method of deep learning music generation creates music, one can not simply compare how close the output is to the input. A perfectly similar output would not be a viable variation on the musical input. Instead, salient features of the music, its musical characteristics are identified and followed through the creation process. In the first stage, these musical characteristics are retained in the automated transcription of MIDI files from audio recordings. In the second stage, these musical characteristics are identifiable through the accuracy of the MIDI transcriptions as they are used for model training. Finally, these musical characteristics are identified in the RNN model generated MIDI files.

5.1 Preparing Music Corpora For Training

Each of the four musical models are composed of a corpus of MIDI files which are created as a representation of audio recordings comprising each model respectively. Because several of the models are created from improvisations, a method of audio to MIDI transcription is necessary since commercially produced MIDI files and scores are not available for these examples. Additionally, because a rather large corpus of MIDI examples are needed for training the RNN, transcription by ear is not realistic. In order to com-
putationally accomplish this task Magenta’s dual objective piano transcriber, Onsets and Frames was used.[18] The preparation for training each model is a three part process. First, audio recordings that represent each model are collected. Next, the collected audio files are transformed into proper lengths for training. Finally, these audio files are converted into MIDI files as a corpus for model training.

Figure 5.1: The preparation for training each model is a three part process. First, audio recordings that represent each model are collected. Next, the collected audio files are transformed into proper lengths for training. Finally, these audio files are converted into MIDI files as a corpus for model training.

The audio data-sets collected for corpus creation are wave recordings of solo piano music. Considerations for data-set requirements include: high fidelity, limited audience noise in live recordings, and perceived validity of data-set uniformity. In some cases audio files that meet the first two criteria are disregarded because they seem out of character for the desired corpus. Each models success is better measured when members of the training corpus contain similar features.

Before the conversion from audio file to MIDI file can be executed, file length must be considered. MIDI file length is an important factor for training models. Some MIDI files are regarded by Magenta as too long to create sequence examples for training while others are too short. Through the process of debugging and attempting to get as many sequence examples for training purposes an audio file length of 60 seconds is found to be most fruitful. Because of this, the audio command line processing library FFMPEG[25] is used to first concatenate all audio files in a directory. Next, FFMPEG is used to cut the
now concatenated audio file into many 60 second audio files.

Once the audio files of each data-set are the appropriate length, it is time to convert audio into MIDI. With Onsets and Frames, polyphonic piano transcription is possible by using a deep convolutional and recurrent neural network which is trained to predict onsets and frames. When this method detects an onset audio event in a recording it will only activate if a pitch relating to the same frame is also detected. This method of transcription is much more accuracy than prior automated piano transcription attempts by the Magenta team and others. Additionally the model can predict relative velocities in audio files, resulting in more natural-sounding MIDI transcriptions.[18]

5.2 Individual Model Training and Evaluation of Model Performance

Each of the four musical models are trained for approximately 60,000 steps with a batch size of 64 and RNN layer sizes of 128, 128, 128. The training time for each model is approximately 48 hours on an eight core NVIDIA TITAN GPU.

5.2.1 $D(u)0$ by Mara Gibson

$D(u)0$ is a three movement piano piece composed for the Bugallo-Williams Piano Duo by composer Mara Gibson. $D(u)0$ is described by the author as an investigation “into the movement from: the mechanical to human (machine to man), the imaginary to real, a music box to performer, and the move from two performers to one.”[16] Programmatically, the work is well suited for an experiment involving composition with machine learning. In terms of style $D(u)0$ features thematic repetition and variation especially in terms of interval content with registrar shifts. In general, $D(u)0$ makes extensive use of the piano’s extreme registers. Much of the work is very high in register and some is very low. At other times the piece makes use of the four hands available in a piano duo by playing extremely high and extremely low simultaneously. Other notable features in the work are repeating and overlapping figures with varying subdivisions of meter, the extended techniques such as “knocking on the piano” and playing inside the piano, repeated notes in the upper register,
and the use of melodic pivot points. For a piano duet piece, D(u)o is often very sparse in density and is at times soft in dynamics.

Figure 5.2 is an example of the published score of $D(u)0$. In this example from the second movement of the score, prominent musical characteristics are present such as extreme register use, repeated notes in the upper register and repeating figures.

Unlike other models to be examined, $D(u)0$ was a score before it was a dataset of audio files performed by the Bugallo-Williams Piano Duo or a transcribed corpus of MIDI files. Because of this distinguishing trait, $D(u)0$ can be used to observe some features of Magenta’s Onsets and Frames, polyphonic piano transcription results in comparison to the score as seen in figure 5.2. Figure 5.3 is the exact two measures shown in the published score example in figure 5.2. However, this score comes from the MIDI corpus transcriptions used to train the model. Note how with Onsets and Frames, the piano transcription rendering differs from the score in metric presentation while retaining metric relationships in general as well as pitch information. In this example the two measure score example in figure 5.2 becomes a three measure example in the audio transcription shown in 5.3. This anomaly can be explained. The corpus of MIDI files is not based on the score. Instead they are a representation of audio recording of human performance of the work. Therefore, it is easy to examine the humanized realization of this score in expressive performance. It’s also important to understand that in this case a piano duet is being transcribed using Magenta’s Onsets and Frames which is trained to transcribe solo piano recordings.

Figure 5.4 shows an annotated example generation created by the D(u)o model. Notice the similar use of extreme register and repeated notes in the upper register. Additionally, this is a short example of a repeating figure.

5.2.2 The Compositions and Improvisations of Thelonious Monk

The piano playing of Thelonious Monk has always been controversial in the world of jazz critics and listeners.\[11\] Unlike his contemporaries in the Bebop era of jazz, Monk did not display normative pianistic virtuosity. Instead he employed the use of space with
Figure 5.2: D(u)o published score example showing extreme register use, repeated notes in the upper register and repeating figures
Figure 5.3: This score excerpt is the exact two measures shown in the published score example in figure 5.2. transcribed by Magenta’s Onsets and Frames rendered from an audio recording of a performance of D(u)o
Figure 5.4: D(u)o generation example with similar use of extreme register and repeated notes in the upper register as well as repeating figures
remarkable intention and elegance. Additionally Monk favored dissonant voicing. Often these voicings employed right hand clusters of notes and an open 5th or 7th in the left hand. Monk made frequent use of melodic figures and chord voicings that placed an emphasis on altered chord extensions such a #11s, #5s, #9s and b9s. When performing solo piano he would sometimes revert to a "stride" style of playing. However, Monk’s stride was sparse in comparison to recordings of the stride pianists such as Art Tatum or James P. Johnson. From every angle Monk’s playing is very unique and recognizable. For this reason his piano playing is well suited for a machine learning model since the effectiveness of this models should be easily evaluated by comparing musical features observed from audio recordings, MIDI corpus examples and MIDI model generations.

Figure 5.4 shows an annotated example from the corpus of MIDI files used to create the Thelonious Monk model. The excerpt chosen is from a solo piano performance of Monk’s tune Monk’s Point. Example of left hand open 7th chord shells can be seen in every measure. In measure 1 a right hand cluster voicing is used making a Bb7(#9) chord. Additional cluster voicings are in every measure except the last. In measure 5 the chord is a Eb7 with a grace note indicating a major 7 with a simultaneously sounding minor seven. One final chord extension alteration can be found in measure 7 with a C7(b9).

Figure 5.5 is an example generation created by the Thelonious Monk model. In general, this example is less harmonically predictable than figure 5.4. In addition it is full of odd spellings of accidentals. However, it does show remarkable similarity in chord structure. In measure 1 an enharmonically spelled Ab7b5 with an open 7th left hand voicing is present. Measure 2 contains a C major7(b9) with both a right hand cluster and open seventh voicing. In measure 6 a clustered Ab minor(5) is presented. Finally, in measure 7 an awkwardly spelled Gb7 with a right hand cluster and left hand open 7th moves to a Bb tritone voicing.

5.2.3 The Piano Playing of James Carroll Booker III

James Carroll Booker III was born in New Orleans, La in 1939.[42] His piano playing is as unique as his life and diverse musical influences. Booker was, by virtue of his location,
Figure 5.5: A transcription of a Monk audio recording showing left hand open 7th chord shells, right hand cluster voicing and altered chord extensions
Figure 5.6: A MIDI generation created by the Monk model showing left hand open 7th chord shells, right hand cluster voicing and altered chord extensions
one of the links in the chain of "New Orleans piano professors" that created rhythm and blues, New Orleans stride, early rock and roll and funk music. He was intimately familiar with the other innovators of that style of piano playing such as Tuts Washington, Huey Smith, Professor Longhair, Fats Domino and his contemporaries, Allen Toussaint and Dr. John. However, what makes his style particularly interesting is the fusion of all this with his early classical musical education and his virtuosic performance abilities of European classical piano literature. Because of this Booker would often interweave boogie-woogie or soul ballad piano stylings with glimpses of a Chopin waltz or a prelude by Rachmaninoff. Booker’s piano style was very dense and often seemed unbelievable because of his amazing technique and large hand size. All of these qualities made booker’s playing extremely unique. As a result he is an interesting candidate for a machine learning model since judging the success of the model’s generations will more easily be evaluated.

In figure 5.7 an example of the corpus used to train the James Booker model is presented. The excerpt come from a solo piano performance of the Booker original song, So Swell When You’re Well, which was a popular hit recording for Fats Domino. In this particular Booker recording employs his trademark boogie-woogie style. This style is very similar to Fats Domino’s boogie style. However, as can be seen in the example he uses very large chords with octave doubling. The division of left hand syncopated sixteenth notes is notable. By comparison, figure 5.8 shows a very similarly boogie-woogie figure which was generated by the Booker model. The only difference worth mentioning is that the generated version in figure 5.8 does not include any triplets. This could be explained since the default generation step operates at the sixteenth note level. Therefore, triplets are far less common.
Figure 5.7: A transcription of an audio recording of James Booker showing large chords with left hand syncopated sixteenth "boogie" figures.
Figure 5.8: A MIDI file generated by the James Booker model showing large chords with left hand syncopated sixteenth "boogie" figures.
5.2.4 Piano Improvisations of William A Thompson IV

The piano playing of the author, William Thompson is in this case, completely improvised. In these examples no plan, preconceived chord structure or melodic material was employed. Each performance was created on the spot with the goal of creating a spontaneous composition. Regardless, there are many common musical methods employed. In general these improvisations imitate various 20th century piano works of the classical tradition combined with modern jazz piano devices. Much of the material is inspired by the compositions of Claude Debussy, Maurice Ravel, Alexander Scriabin, Sergei Prokofiev, Bill Evans and Thelonious Monk to name a few. Some common devises used are chromatic melodic material, moving inner voices, moving parallel structures, contrary motion, use of pedal points, extensive use of the sustain pedal, altered extended harmony, rhythmically syncopated contrapuntal like figures and dense harmonic structures.

Figure 5.9 is an example from the corpus used to train the William Thompson model. Because of the overlap of material in register, this example shows each measure in four staves. In it ample use of the sustain pedal can me seen though the many tied notes created by the Onsets and Frames transcription model. Additional this excerpt contains examples of chromatic melodic material in measure 2, 3 and 4. Inner voice movement can be found in measures 2 through 5. One pedal point in the lowest voice is demonstrated in the last 4 measures.

In figure 5.10 a MIDI file generated by the William Thompson model is examined that shows many similarities to the corpus example. Extensive sustain pedal use is evident due to the many tied notes. Chromatic melodic material is present in measures 1, 2 and 6. Pedal points in various registers can be seen in measures 3, 4, 5, 6, and 7. Inner voice movement is prominent in measures 4 and 6.

Evaluation of musical models can be achieved though listening and describing musical characteristics. However, identification of features which are important for speech models are not as easily understood.
Figure 5.9: A MIDI transcription of and audio recording by William Thompson showing chromatic material, moving inner voices, and pedal points.
Figure 5.10: A MIDI generation created by the William Thompson model showing chromatic material, moving inner voices, and pedal points.
Chapter 6
Models Created Representing Characteristics of Speech

Machine learning models based on characteristics of speech are created through a similar process as models based on music. These speech models are made up of MIDI transcriptions of audio recordings of human speech. Collections of these transcriptions form corpora that when trained into machine learning models, emulate the sonic characteristics of specific emotional states of speech. This compositional process is comprised of three emotional states of speech: neutral speech, angry speech and amused speech. Each of these three emotional states is used to create a model intended for human pianist performance and a model intended for Yamaha Disklavier machine performance. The Yamaha Disklavier piano is a modern player piano that can read MIDI files. A total of six speech models are created to account for human and machine performance options representing each of the three selected emotional state of speech.

6.1 Preparing Speech Corpora For Training

Each of the six speech models are created from a corpus of MIDI files. These MIDI files are automatically transcribed from audio recordings of voice actors reading text spoken with the intent of emulating specific emotional moods of speech. The source of these recordings is the The Emotional Voices Database: Towards Controlling the Emotional Expressiveness in Voice Generation Systems, or EmoV-DB for short. EmoV-DB includes recordings of two males speakers and two female speakers. Each recorded speaker seeks to replicate the emotional styles of a neutral, sleepy, angry, disgusted or amused state. The original purpose of EmoV-DB is to aid in the field of emotional speech synthesis. The emotional styles of neutral, amused and angry are included in this compositional process.

The neutral, amused and angry speech datasets of audio files are converted to MIDI files for model training through a similar process as the music models. Due to the complexity of sonic characteristics of recorded speech and the necessity of a large corpus for training
models, transcription by ear is not practical. In order to computationally accomplish this task two separate transcription methods are used with two distinct goals. One transcription method is used to create MIDI corpora intended for machine performance, and another transcription method is used to create MIDI corpora that are performable by a human pianist.

To computationally transcribe audio files into a MIDI file corpora intended for machine performance, the command line tool, WaoN[20] is used to extract spectra from recorded speech audio files into MIDI files that contain polyphonic pitch information. The result is often a MIDI file that can contain very many notes based on an audio file’s spectral content. When applied to speech recordings, WaoN can detect enough pitch information to make MIDI representations of speech recordings that are almost, if not actually, intelligible as specific words by the listener. Listening to these MIDI files is very similar to hearing the ”speaking piano” works of Peter Ablinger.[3] This transcription method is ideal for machine performance by a player piano since these machine performances are not restricted to any range of possible simultaneously sounding pitches. However, because of the abundance of pitch information, these MIDI speech realizations are in most cases not performable by a human pianist.

To form the human performable MIDI corpora, Magenta’s dual objective piano transcriber, Onsets and Frames is used. This transcription method is intended for piano transcription. However, when applied to human speech recordings, Onsets and Frames creates MIDI files that contain one to six simultaneous pitches on average, which are usually practical for human pianist performances.

Much like process of preparation for training music models, the speech models for both human and machine performance are prepared in a three part process. First, audio recordings that represent each model are collected. Next, the collected audio files are transformed into proper lengths for training. A speech audio file length of 60 seconds is found to be most fruitful. Because of this, the audio command line processing library
FFMPEG is used to first concatenate all audio files in a directory. Next, FFMPEG is used to cut the now concatenated audio file into many 60 second audio files. Finally, these audio files are converted into MIDI files as a corpus for model training.

Figure 6.1: Speech models for both human and machine performance are prepared in a three part process. First, audio recordings that represent each model are collected. Next, the collected audio files are transformed into proper lengths for training. Finally, these audio files are converted into MIDI files as a corpus for model training.

6.2 Individual Model Training and Evaluation of Model Performance

Each of the three speech models are trained for approximately 60,000 steps with a batch size of 64 and RNN layer sizes of 128, 128, 128. The training time for each model is approximately 48 hours on an eight core NVIDIA TITAN GPU.

6.2.1 Speech Emotion Recognition Evaluation

Evaluation of speech models is important to demonstrate that the MIDI corpora used in training and the sequences generated by the models retain the audio features that are important for emotional recognition of speech.

Speech is a very fast and accurate method of communication amongst humans. Additionally, speech is an efficient method of human and machine interaction. When humans
communicate with other humans, the clarity of a received message is amplified though the use of all available senses in a manor that allows for the understanding of literal words as well as the emotional state of the speaker. This aspect of communication through language is natural and easy for human listeners. However, this is much more difficult for a machine listener. Because of this discrepancy, the evolving practice of speech emotion recognition (SER) seeks to understand and use emotional speech knowledge to improve human to machine communication. To achieve this SER needs to recognize the emotional aspects of speech separated from semantic content.[13] The task of SER is difficult for several reasons. To classify SER it is necessary to extract audio features from speech and associate the behavior of specific features with specific emotional states. This association is a challenge to make because of factors such as varying: speaking styles, rates of speech, dialects, and cultural considerations. The most effective modern methods of SER depend on the use of Deep Learning (DL) techniques. DL methods allow for raw data input with classification output. The most challenging hurdle facing SER through DL is the relatively small amounts of emotionally labeled speech datasets.[24]

MIDI was not designed to express emotional states of speech. In light of this, SER can be useful in validating that emotional states of speech are retained through the transcription process. SER only functions by analysing audio files. Because of this, MIDI transcriptions are converted into audio file format piano renderings for analysis. Similarly, MIDI generations created by these models are converted to audio files for analysis.

6.2.2 Vokaturi and Emotion Recognition

The SER software, Vokaturi, is employed to each speech model to evaluate the accuracy of emotional categorisation of the complete datasets of audio recordings representing neutral, amused and angry speech, the corpora of transcribed MIDI files used for training, and a collection generated MIDI files from each machine learning model. Collected model generations are ten eight measure generations per model.

The Vokaturi software’s algorithms are designed and maintained by Paul Boersma,
professor of Phonetic Sciences at the University of Amsterdam.[5] In addition, Boersma is the main author of the world’s leading speech analysis software Praat. OpenVokaturi is the open-source version of the Vokaturi library which is employed with Python. These Python scripts can be used to measure the average emotion probabilities in an existing WAV file. When executed, these scripts print one normalized floating point number for the percentage of the following emotions: Neutral, Happy, Sad, Angry and Fear. The Cross-validated accuracy of the five emotion classification is 66.5 percent.

6.2.3 Neutral Speech Models

Figure 6.2 is a graph showing the results of using Vokaturi to classify the neutral speech database audio recordings, the corpus for training the machine performance model, the corpus for training the human performance model. The analysis of the Dataset predicted only a neutral emotion. Similarly, the Vokaturi classification of audio files created from the corpus of MIDI files used to train the neutral speech model for machine performance predicted only a neutral emotion. The results of using Vokaturi classify audio files created from the corpus of MIDI files used to train the neutral speech model for human performance contain very minor deviation from the accurate dataset and machine performance corpus classifications. A very small amount of the classification was labeled as sad. In figure 6.3 similar classifications are presented in the evaluation of files generated by the the neutral speech for machine performance trained model. The classification of over 0.8 for neutral far overshadows the sad and fearful classifications. The same can not be said for generation produced by the neutral speech model for human performance. It’s classification has no indication of neutrality and instead is mostly labeled as angry with some sadness.
Figure 6.2: Corpus for Machine Performance, Corpus for Human Performance. The analysis of the Dataset predicted only a neutral emotion. The corpus of MIDI files used to train the neutral speech model for machine performance predicted only a neutral emotion. The corpus of MIDI files used to train the neutral speech model for human performance contain very minor deviation.

Figure 6.3: The model generations neutral speech for machine performance is primarily accurate. The model for human performance on the other hand show only an angry classification.
6.2.4 Amused Speech Models

The Vokaturi classification for amused speech is much less predictable than classifications of neutral speech. Perhaps this is because Vokaturi can predict emotions that are either neutral, happy, sad, angry or fearful. Although amusement seems to be most similar to happiness, this is too big of an assumption to make. The dataset examples of amused speech are primarily composed of people laughing. It’s possible that laughter can be thought of as a much more excited form of speech than happiness. This is a possible explanation for these Vokaturi results which include labels of the excited speech states of anger and sadness. The Vokaturi audio dataset analyze results in a measurement of 0.3 for happiness and just under 0.7 for anger. When this result is compared to analysis of the MIDI corpus of the amused speech for machine performance there is only similarity in the anger and happiness. Anger in this corpus is close to the dataset at 0.5 while happiness is evaluated much lower at well under 0.1. Instead, the amused speech for machine performance corpus resulted in a high sadness evaluation between 0.4 and 0.5. The MIDI corpus of the amused speech for human performance is identical to the audio dataset. Generation of models trained on corpora of amused speech are also not consistent. The amused model for machine performance generations is entirely classified as sad, while the model generations for human performance are entirely angry. Perhaps there is some link to the training data since anger and sadness are both identified in the dataset analysis.

![Amused speech classifications are not consistent.](image)

Figure 6.4: Amused speech classifications are not consistent.
6.2.5 Angry Speech Models

the Vokaturi evaluation of the Angry speech audio dataset is almost entirely classified as angry. However, it does show trace amounts of both neutral and happy evaluations. The MIDI corpus of angry speech for machine performance also shows a classification of a rating of approximately 0.3 for sadness and a very small rating for happiness. The MIDI corpus for human performance evaluation was almost the same as the dataset analysis. The human performance corpus only differed from the dataset in it’s lack of neutral classification.

Figure 6.5: Amused speech classifications are not consistent.

Figure 6.6: The Angry speech audio dataset and Corpora are primarily accurate.
Generations created by angry speech models for both machine and human performance are labeled as only angry.

**Figure 6.7:** The Angry speech generations are overwhelmingly accurate.
Chapter 7
The Compositional Process Through Computational Steering

7.1 Cross Synthesizing Models

Computational steering is the interactive control over a computational process through adjustment of input parameters. [31] Each composition is created by adjusting parameters in the python script prior to generation. The resulting generation of MIDI data is realized as the score to be performed. This compositional process allows for model based cross synthesis through computational steering. Model based cross synthesis is a term used to describe the process of having one machine learning model generate material based on material that a separate model has either been trained on or has produced in generation.

Magenta is capable of setting a primer MIDI file before each generation. When set, the model will attempt to create something similar to the primer. Normative Magenta score generation using primer MIDI files can result in the creation of new MIDI files that continue and elaborate homogeneously on primer MIDI data. The pre-trained Polyphony RNN model is trained on a corpus of Bach chorales. Using this model, a primer file that contains four part counter-point can easily result in a homogeneous generation that includes the injected primer and the following newly generated MIDI data. Magenta’s pre-trained model is very good at this. The model configures itself around features of the primer and creates new MIDI data based on what it has learned from it’s training process. However, Models in this compositional process are asked to generate new MIDI data based on MIDI primers that often do not resemble MIDI corpus upon which they have trained. In this process, musical models are asked to create music based on speech primers and speech models are asked to create speech based on musical primers.

7.2 Music Inspired by Speech and Speech Inspired by Music

One of the most important techniques employed in this compositional process generates speech like material inspired by music. Additionally, this process can generate musical ma-
material that is inspired by speech. This is an example of model based cross synthesis in which a model is generating material based on the training data for a separate model. This is possible because of the way that Magenta primers behave during generation. Examination of an example of this process is beneficial. In the following example only one sequence is generated for the sake of clarity. This generation is produced by a musical model that has been primed by a MIDI file that approximates human speech. This generation calls on the bundle file, booker.mag which contains checkpoint information from the model trained on the piano playing of James Booker intended for machine performance. The primer used in this generation is taken from the corpus upon which the neutral speech model was trained. As a result, it contains MIDI data that resembles spectral of speech in a neutral mood. The generation is set to condition on primer but it is not set to inject the primer. As a result, the score shows the primer file in the generation before the newly generated material and not at the same time.

```
generate(
    bundle_name="booker.mag",
    sequence_generator=polyphony_sequence_generator,
    generator_id="polyphony",
    midi_filename="neutral-booker_4bar.mid",
    total_length_steps=64,
    condition_on_primer=True,
    inject_primer_during_generation=False,
    temperature=1.0,
    primer_filename="bea_neutral_1bar.mid"
)
```

The generated score in figure 7.1 shows a MIDI approximation of speech spectra in the first measure. In the second measure there is a lot of MIDI data that looks and sounds very similar to the speech spectra. However, simultaneously, musical elements begin to appear since the Booker model is creating all new MIDI data. The most noticeable musical
Figure 7.1: Speech Primer with Musical Model
features consistent with the Booker models training appear in the bass clef. Rhythmically, measures two, three and four have the characteristics of a shuffle or likely "boogie-woogie" pattern. Measure two seems to be centered around A in the left hand while in measure three F appears several times. Melodically there seems to be a bass-line like structure with some often repeated pitches. These features are consistent with the Booker model. Harmonic structures also appear in the final three measures as thirds and fourths are much more common than in the speech spectra of the first measure. In some instances entire triads are present which are very consistent with expected Booker-like content. As this example progresses the presence of functional harmony is more clear. The final measure contains the following progression: E minor 7, E7 9, Ab augmented, E minor, E minor/B, and G major 7. Through the examination of this example it seems that the primer file, which contrasts the model, is most potent in influence at the beginning of the generation. Steps generated further from the site of primer injection seem to resemble model features more than those inspired by primer files.
In this next generation an example of a speech model primed by a musical MIDI file is observed. Here the bundle file, angry_OnF.mag is called on to generate new MIDI data. This file contains checkpoint information that is the result of training on MIDI representations of angry speech for human performance. The primer used is a MIDI transcription of three measures of Thelonious Monk playing *Nice Work If You Can Get It*. Like the previous example, this generation is conditioned on the primer. In addition, the primer is injected into the new MIDI generation. MIDI data seems to more resemble characteristics of the model as time progresses in the generation.

```python
generate(
    bundle_name="angry_OnF.mag",
    sequence_generator=polyplyphony_sequence_generator,
    generator_id="polyplyphony",
    midi_filename="monk2angry_8bar.mid",
    total_length_steps=96,
    condition_on_primer=True,
    inject_primer_during_generation=True,
    temperature=1.0,
    primer_filename="monk_2bar.mid")
```

As can be seen in the score displayed in figure 7.2, the first three bars are made up by the primer. In the following three measures the primer is again present but surrounded by new MIDI data generated by the angry speech model. In the final measure the primer injection is no longer present. It’s much more difficult to understand how speech models react to music primers since speech spectra is not normally analyzed in musical notation.

In addition to conditioning and injecting music and speech on to one another, new generations can be created by one model are used as primers for other models. This cross model synthesis is explored through a new process referred to as successive primers.
Figure 7.2: Speech Primer with Musical Model
7.3 Successive Primers

The use of successive primers is a new novel method of composing music with multiple primers and is an example of model based cross synthesis in which a model generates material based on MIDI data generated by another model. This is the major computational steering technique used in this compositional process. Successive primers work by sequentially looping multiple MIDI file generations and resulting in one fully realized composition. First, an initial primer file is assigned for the first model generation. The first model generation then becomes the primer file for the second generation. The second generated MIDI file then becomes the primer for the third generation. This process is repeated as a for loop in the python script until the desired number of successive generation are completed resulting in one final composition. Each new generation is influenced by previous generation either by primer injection, primer conditioning or both. The use of successive primers allows for the use of multiple models that influence the outcome of one final generation. This process is referred to as model based cross synthesis because each generation, except the first generation is reinterpreting data produced by another model. Additionally, the use of successive primers allows for the creation of one score that is the product of many different parameters that change throughout the generation. In normal single file generations parameters do not interpolate and are set to one condition before generation occurs. Each link in the successive generation chain can be set to a different number of measures, a different number of steps per measure, different models and different temperatures of randomness. Also, each link can be set to true or false for both conditioning on it’s primer and the injection of it’s primer. Examination of a specific example in the form of Etude III. 1712 Steam Engine (for human performer) can illustrate these principals well since it’s programming is more simple than other etudes. The complete code can be found in appendix B.3 and the complete score is accessible in appendix A.3.

This code generates only three sequences with one initial primer and two successive primers. The final output is one MIDI file that includes all previous steps as realized by
the final model generation named "III.human_musicPrimer_wativ_slow18bars3.mid".

bundles = ("neutral_onF.mag", "booker.mag", "wativ.mag")
filenames = ["primer.mid", "primer1.mid",
            "III.human_musicPrimer_wativ_slow18bars3.mid"]
steps = []
bars = [20, 8, 8]
s_p_q = [1, 2, 2]
totalBars = 0
for idx, x in enumerate(bars):
    totalBars = totalBars + x
    steps.append(totalBars * s_p_q[idx] * 4)
    print(totalBars)
conditions = [True, True, True]
injections = [True, True, False]
temperatures = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0]
primers = ["wativ_slow18bars.mid", "primer.mid", "primer1.mid"]

The initial generation in this short three generation composition uses the neutral speech for human performance model, is twenty measures in length, and is set to one step per quarter. The setting of one step per quarter means that the smallest possible rhythmic unit in generation is a quarter note. This is represented in the code as s_p_q. Because this is the first of three successive primers, this generation uses a primer file that is not generated in this sequence. The primer file, "wativ_slow18bars.mid", is an eighteen measure piano improvisation and is one MIDI file included in the MIDI corpus used to train the William Thompson model. This generation is conditioned on the primer and injected with the primer. This is clearly seen in the score as the first eighteen measures include the primer file realized by the neutral speech for human performance model as seen in figure 7.4. The remaining two measures of the generation are completed by the model attempting
to continue the injection with what it has learned from it’s training on neutral speech.

The second generation and first successive primer makes use of the James Booker model. It’s primer input is the generated MIDI output from the first generation. In this second generation eight new steps are added making the total length of the generation now twenty-eight measures. The setting of two step per quarter means that the smallest possible rhythmic unit in generation is an eighth note. Again this generation is conditioned on and injected with the primer. In measure twenty-one of Etude III. the material from the second primer is seen and is now reinterpreted with 8th subdivisions and James Booker model inspired embellishment.

The third and final generation for Etude III. uses the William Thompson model and takes the twenty-eight measures generated in the second generation as it’s primer. Here eight more measures are added to generate a total output of thirty-six measures. This generation is conditioned on it’s primer but the primer is not injected. Because of this the final eight measure of the etude are inspired by previous generation. However, previously generated material is not recounted exactly.
Figure 7.3: Successive Primers Process for Etude III.
Figure 7.4: Etude III. Annotated Score
7.4 Seven Piano Etudes Speaks the Moody Machine, Compositional Observations

7.4.1 Program Notes

Seven Piano Etudes Speaks the Moody Machine was created via machine learning with the goal of hybridizing the musical qualities of three different emotional states of human speech with four different styles of piano music. Through this process, amused, angry or neutral speech have been melded with piano music created by Thelonious Monk, James Booker, Mara Gibson and William Thompson. These etudes are intended to be performed on one Yamaha Disklavier piano with two performers, one human pianist and one machine player piano. The Etudes that are intended to be performed by a human pianist are titled after notable events from the industrial revolution while those intended for machine performance are titled after possible future events regarding the rise of artificial intelligence as predicted by the author Ray Kurzweil. In the final etude both human and machine perform together representing the singularity. The contrast between events in the industrial revolution and future events is intended to entice listeners to reconsider each from different perspectives of time.

Seven Piano Etudes Speaks the Moody Machine

I. 1440 Type (Human Performer)
II. 2029 Claim of Consciousness (Machine Performer)
III. 1712 Steam Engine (Human Performer)
IV. 2030 Mind Upload (Machine Performer)
V. 1844 Telegraph (Human Performer)
VI. 2040 Full Immersion (Machine Performer)
VII. 2045 The Singularity (Human and Machine Performer)
7.4.2 Analysis

Each of these etudes are created with one generation that is the result of the successive primer process. Etudes intended for machine performance are not edited with the exception of added dynamic markings. Scores intended for human performance are edited in an effort to make them legible and physically performable by a human pianist. Dynamic markings are added by hand to both machine and human performed etudes. The entirety of the generated score can be found in appendix A. Additionally, all python scripts used to create these scores is available for review in appendix B.

Three distinct categories of generations are present in these etudes: those created for human performance, those created for machine performance and those created for simultaneous machine and human performance. A detailed examination of one etude from each of these three categories is discussed in an effort to better understand the compositional process and the musical interest created.

Etude V. 1844 Telegraph (Human Performer) is an interesting piece intended for human performance. The python script parameters that create this piece are listed here.

```python
bundles = ("wativ.mag", "neutral_OnF.mag", "duo.mag", "wativ.mag",
"monk_60sec.mag", "amused_OnF.mag", "monk_60sec.mag")
filenames = ["primer.mid", "primer1.mid", "primer2.mid", "primer3.mid",
"primer4.mid", "primer5.mid", "V.human_music_wativ_drone_12bar7.mid"]
steps = []
bars = [16, 2, 8, 12, 2, 4, 4]
s_p_q = [4, 1, 4, 8, 2, 4, 1]
totalBars = 0
for idx, x in enumerate(bars):
    totalBars = totalBars + x
    steps.append(totalBars * s_p_q[idx] * 4)
print(idx)
```

The musical interest in etude V. is in its use of theme and variation. Through the sequential primer process musical material is presented and then reintroduced several times in the piece. Each time material is sequentially present in the score it is altered through the manipulation of length, rhythmic scale and model interpretation. These manipulations are cumulative in the sequential primers process. If a series of sequential primers is injected five times its contents can be altered drastically since its final generation has undergone five different transformations. Additionally each generation’s newly added material becomes part of the injection and conditioning for the next generation. Etude V. has some great examples of this. As indicated in the code there are five total primer injections in this piece. The initial generation is the injection of the first primer, wativ_drone_12bar.mid. This primer is a piano improvisation by the William Thompson. The next injection occurs at measure nineteen. Here the model being primed is one trained on neutral speech. Because the model has been asked to produce MIDI data with the largest subdivision of sixteenth notes ($s_{p,q} = 4$), the reinterpretation of the initial generation is injected with new rhythmic scale with double the subdivision level. The original primer content is realized with sixteenth note possibilities. Here the model is not conditioning the generation on its primer. Therefore, it is generating new material around the primer. Things start to get interesting in the next four generations. Each subsequent injection uses newly generated material which was produced in past generations as its primer. As a result the fifth and final generation is linked to the first injection. However, it does not resemble it. Instead it is the product of distant variation. An examination of injection tracing can be seen in figure 7.5. Parts of the score circled in red indicate material form the primer file injected.
into later successive generations. Yellow indicates material that first appeared in the first injection. Blue indicates the second injection while green and purple represent four and five. It’s interesting to examine how material changes from one generation to the next.

Figure 7.5: Parts of the score circled in red indicate material form the primer file injected into later successive generations. Yellow indicates material that first appeared in the first injection. Blue indicates the second injection while green and purple represent four and five.
Etude IV. 2030 Mind Upload (Machine Performer) takes its title from author Kurzweil’s book The Singularity Is Near. According to Kurzweil’s prediction in the year 2030 humans will perfect mind uploading which will make human minds software based. The composition is meant to capture this idea by shifting through the human experience of speech at inhuman speeds. Etude IV. is diverse in density while homogeneous in texture through its use of successive primer generation. The following are the parameters that create this score.

bundles = ("wativ.mag", "angry.mag", "booker.mag", "wativ.mag", "duo.mag", "wativ.mag", "duo.mag")
filenames = ["primer.mid", "primer1.mid", "primer2.mid", "primer3.mid", "primer4.mid", "primer5.mid", "IV.machine_speechPrimer_sam_angry_7bar_200bpm2.mid"]
steps = []
bars = [16, 4, 4, 4, 4, 4, 4]
s_p_q = [4, 32, 1, 16, 2, 32, 1]
totalBars = 0
for idx, x in enumerate(bars):
    totalBars = totalBars + x
    steps.append(totalBars * s_p_q[idx] * 4)
    print(idx)
conditions = [False, False, False, False, False, False, False]
injections = [True, False, True, False, True, False, False]
temperatures = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0]
primers = ["sam_angry_7bar_200bpm.mid", "primer.mid", "primer1.mid", "primer2.mid", "primer3.mid", "primer4.mid", "primer5.mid"]
Etude IV. makes good use of the self playing piano since it’s performance would not be feasible by a human pianist. This is due to it’s extremely dense sections and the presence of one hundred twenty-eighth notes at a tempo of two hundred beats per minute. Etude IV. does not condition any generations based on a primer. Instead, this etude creates musical interest by contrasting sections that result from injection points and rhythmic density choices set before generation. Etude IV. is much more spectral in it’s imitation of speech than other etudes in this collection. As a result it has very little harmonically identifiable moments. Even though only one successive primer generation uses an angry speech model. Most of the piece sounds like angry speech. This is accomplished through a series short generations and injections of speech like material. The first generation is sixteen measures in length. The remaining six generations are all four measures in length. This regularity in sequential primer length allows for greater clarity in the articulation of parameters for each generation. The first generation is injected with a MIDI file that is a part of the angry speech model. It is employed because it’s spectral MIDI representation is an effective example of angry speech. Additionally, the first sixteen measure generation creates using the William Thompson model. However, it is difficult to aurally or visually distinguish this models presence since the primer injection takes up all but two measure of the sequence. Figure 7.6 shows the speech primer in score representation.

The initial injection differs from the primer in pitch content only by addition. In measure eight the same pitch structure is present but surrounded by new pitches. As a result the injection sounds like angry speech and even more dense. The second generation does not inject the primer. However, it does retain the spectral qualities of angry speech by calling on the angry speech model to generate. Here things get interesting from a rhythmic standpoint. The steps per quarter note is set to 32, meaning that in this generation subdivisions as small as a 128th are possible. For this reason, several dectuplets are present. Some measures, such as measure nineteen, are so dense that one measure takes four systems to be realized in the score. This section is effective in created a fast and angry speech
Figure 7.6: Etude IV. Angry Speech Spectra Primer
imitation. To contrast this, the third successive primer generation is set so that it’s smallest possible rhythmic subdivision is a quarter note. Interestingly, this generation also injects the former generation. In this sequence the initially injected primer is seen now in dense cluster chords at the rate of four to a measure. This does not sound like speech. However, it is easy to see and hear that it uses speech spectra as a source to create musical interest. The fact that the James Booker model is used is almost lost with the combination of rhythmic restriction and primer injection. The fourth generation at measure twenty-five is the first segment that sound as if it is based on any type of tonal harmony. This is because the William Thompson model is generating without injection. However, four measures later the speech primer is injected again using the D(u)o model at the eight-note rate. Finally, in the sixth generation the finale is presented as the William Thompson model generates a very dense sequence that allows for thirty-two steps per quarter note. To compose an ending the final generation uses the D(u)o model at the quarter note rate. One of the more interesting aspects to observe in the score of Etude IV. is the injection of the same primer material with different models and differing rhythmic density. Figure 7.7 compares each injection in Etude IV.

Etude VII. 2045 The Singularity(Human and Machine Performer) is unique among these pieces. Programmaticly, it represents the Singularity which is a predicted moment in history in which technological growth becomes uncontrollable resulting in unimaginable changes for human kind. It is created to be performed by a Disklavier Piano and a human pianist simultaneously on the same piano. Because of this there could be no registrar overlap between the keys being pressed by the pianist and the keys activated by the machine. This possibility was realized by injecting the primer. In performance the pianist only plays these injected parts while the Disklavier performs only the model generated parts. Both parts were created from group of sequential primers. The machine performed parts never overlap the injection. Because of the repetitive nature of the pianist’s part, all the ferocity of this final etude is machine driven. The density of the machine part is intended to
The following is the code that creates this final etude.

```
bundles = ("angry.mag", "angry.mag", "angry.mag",
"wativ.mag", "wativ.mag", "neutral.mag", "wativ.mag",
"booker.mag", "duo.mag", "wativ.mag")
filenames = ["primer.mid", "primer1.mid", "primer2.mid",
"primer3.mid", "primer4.mid", "primer5.mid", "primer6.mid",
"primer7.mid", "primer8.mid", "VIII.machine_music_wativ_img_2440_9.mid"]
steps = []
bars = [7, 5, 7, 1, 8, 4, 1, 4, 2, 3]
s_p_q = [4, 12, 32, 32, 32, 6, 6, 2, 1, 4]
totalBars = 0
for idx, x in enumerate(bars):
    totalBars = totalBars + x
    steps.append(totalBars * s_p_q[idx] * 4)
```

Figure 7.7: Etude IV. Injection Comparison

represent the predicted lack of human control in the idea of the singularity. The following is the code that creates this final etude.
print(totalBars)

conditions = [False, False, False, True, True, False, True, False, True, False, True, False, True]

injections = [True, True, True, True, False, True, True, True, False, True, False]

temperatures = [1.0, 1.0, 1.0, 1.0, 1.0, 1.4, 1.0, 1.0, 1.0, 1.0]


In total, Etude VII makes use of ten generations as sequential primers and one final phrase which was not composed through machine learning. The generated phrases in this etude change swiftly and are odd lengths. As a result, listening to this etude or reading the score is easiest by following the original primer’s injection. In this case the primer is a piano improvisation by William A Thompson which continues to surface in the composition for various lengths and with varying generated content. The first generation injects the primer. Because the primers length is the same length as the generation, the initial generation is simply the primer with no additional material. Therefore the first seven measures of the piece are performed by the pianist alone.

As the piece continues the machine performer adds more and more cacophonous material which at times effectively hides the human performed theme. The second generation once again injects the primer. However, this time the pianist hands are surrounded by MIDI data generated by the William Thompson model at a sixteenth note rate. Because this sequence is only five measures the theme is left incomplete before it is once again injected in the third sequence. This time the theme is complete. However, it is now surrounded by a lot of new material generated by the angry speech model.

Sequence four begins the theme again but for only one measure when it is disrupted by the presence of sequence five. This generation is the only significant sequence that does
Figure 7.8: The primer is a piano improvisation by William A Thompson which continues to surface in the composition for various lengths and with varying generated content.

Figure 7.9: The beginning of the theme seen in fig 7.8 surrounded by angry speech model material.
not inject the primer. As a result, the William Thompson model generates freely while the human pianist rests for eight bars. This sequence is particularly interesting since the tonal nature of the William Thompson model is at times very recognizable. Additionally, this sequence is very fast with potential subdivisions of 128th notes allowed by the code setting of 32 steps per quarter.

![Figure 7.10](image1.png)

**Figure 7.10:** The Thompson model sounds tonal at times here and very fast.

Generations six, seven and eight once again inject the primer. In generation six the theme is present in its entirety with obvious speech generated material from the neutral speech model. However, this speech is not easily aurally analyzed as neutral because of the speed and micro subdivisions. In sequence seven only one bar of the theme appears before sequence eight presents the theme for one final time with the James Booker model. The Booker model’s presence is recognizable by the left hand broken boogie-woogie pattern.

![Figure 7.11](image2.png)

**Figure 7.11:** The Booker model’s presence is recognizable by the left hand broken boogie-woogie pattern.
The ninth generation does not inject the primer and is performed by the player piano. It’s contents resemble the D(u)o model at the quarter note rate. The final seven measures of the piece and the etudes as a whole was composed by hand and without the aid of machine learning. The right hand is performed by the pianist and the left hand is performed by the machine performer. It is a portion of the original theme.

Figure 7.12: The final four measures of the piece are composed by hand and without the aid of machine learning. The right hand is performed by the pianist and the left hand is performed by the machine performer. A fragment of the original theme, it also appears starting on beat four of m.5 of fig 7.8
Chapter 8
Conclusions

Speech and music are linked in origin. Scholars have speculated regarding the exact condition of their unique relationship. Regardless, the bond between these two methods of human communication is widely accepted. This union of speech and music is justification for new compositional practices that exploit this idea. The compositional goal presented in this research attempts to make meaningful connections regarding human expression by finding a sonically interesting common ground between speech and music.

The creation of the piano etudes that resulted from this research do make for an interesting merging of music and speech. Speech-like symbolic data in the form of MIDI files does make sense as musical content because it can be perceived as and inspired by music. Additionally, music that is inspired by speech does sound musical while retaining speech like qualities. In both cases the term “inspiration” is meant to reference the compositional processes of model based cross synthesis. In this process primers ”inspire” a model to elaborate on foreign MIDI data through the lens of it’s own training data. MIDI data is not intended to imitate speech. In spite of this, MIDI does effectively capture speech spectra in a manor that is practical for musical score representation.

In many situations, composers draw inspiration from unknown resources. This is not entirely different from the processes outlined in this compositional method. Through machine learning composers can employ models that replicate the musical features desired by the composer. In spite of this, the training process occurs with little to no knowledge of what a specific machine learning model values and is learning from. On the other hand, what is unknown to the composer does not necessarily hinder good musical composition. Machine learning models are capable of learning and creating new musical grammar that a human might never perceive. This is difficult to discuss since we can not discuss what we don’t know. However, complete understanding is not needed if machine learning results
are interesting from a musical perspective. These techniques are a means and not an end.

Compositional tools that can access illusive musical features are especially advantageous when attempting to imitate the idiosyncratic emotional expression found in speech. Machine learning models created in this compositional process are effective in producing MIDI generations that resemble various emotional states of speech. Emotional states of speech such as anger, amusement and neutrality can be very complex in description. However difficult they may be to define, they are often easily discernible by the human ear. Human listening is by far the most effective way to evaluate models imitating emotional speech.

The challenge facing the artist utilizing this type of creative process is in the evaluation of generational output. There is a sweet spot that exists somewhere between models that generate new material that is almost identical to it’s training data and models that generate new material that does not resemble it’s training data at all. This is where interesting art can be found. Machines allow humans to create art that is not possible without their aid. However, human perception is the most effective measure of machine success.
Appendix A

Seven Piano Etudes Speaks the Moody Machine, Full Score

Seven Piano Etudes Speaks the Moody Machine was created via machine learning with the goal of hybridizing the musical qualities of three different emotional states of human speech with four different styles of piano music. Through this process amused, angry or neutral speech has been melded with piano music created by Thelonious Monk, James Booker, Mara Gibson and William Thompson. These etudes are intended to be performed on one Yamaha Disklavier piano with two performers, one human pianist and one machine player piano. The Etudes intended to be performed by a human pianist are titled after notable events from the industrial revolution while those intended for machine performance are titled after future events regarding the rise of artificial intelligence as predicted by the author Ray Kurzweil. In the final etude both perform together representing the singularity. The contrast between events in the industrial revolution and future events is intended to entice listeners to reconsider each from different perspectives of time.

I. 1440 Type (for human performer)
II. 2029 Claim of Consciousness (for machine performer)
III. 1712 Steam Engine (for human performer)
IV. 2030 Mind Upload (for machine performer)
V. 1844 Telegraph (for human performer)
VI. 2040 Full Immersion (for machine performer)
VII. 2045 The Singularity (for human and machine performer)
I. 1440 Type (for human performer)
II. 2029 Claim of Consciousness (for machine performer)
III. 1712 Steam Engine (for human performer)
IV. 2030 Mind Upload (for machine performer)
V. 1844 Telegraph (for human performer)
VI. 2040 Full Immersion (for machine performer)
VII. The Singularity (for human performer)

expressive and connected

\[ \text{MIDI notes and symbols here} \]
VII. The Singularity (for machine performer)
Appendix B

Code Corresponding with Etude Generation

B.1 Etude I. 1440 Type (for human performer)

bundles = ("booker.mag", "booker.mag", "amused_OnF.mag", "monk_60sec.mag",
"duo.mag", "wativ.mag")
filenames = ["primer.mid", "primer1.mid", "primer2.mid",
"primer3.mid", "primer4.mid",
"I.human_speechPrimer_bea_neutral_50bpm_2bars_OnF6.mid"]
steps = []
bars = [6, 2, 4, 4, 2, 1]
s_p_q = [4, 4, 8, 4, 8, 8]
totalBars = 0
for idx, x in enumerate(bars):
    totalBars = totalBars + x
    steps.append(totalBars * s_p_q[idx] * 4)
    print(totalBars)
conditions = [True, False, False, True, True, True]
injections = [False, False, False, True, False, False]
temperatures = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0]
primers = ["bea_neutral_50bpm_2bars_OnF.mid", "primer.mid",
"primer1.mid", "primer2.mid", "primer3.mid", "primer4.mid"]
B.2 Etude II. 2029 Claim of Consciousness (for machine performer)

bundles = ("angry.mag", "booker.mag", "wativ.mag", "monk_60sec.mag",
           "angry.mag", "duo.mag")
filenames = ["primer.mid", "primer1.mid", "primer2.mid", "primer3.mid",
            "primer4.mid", "II.machine_musicPrimer_wativ_arp_187bpm_14bar.mid1"]
steps = []
bars = [24, 24, 8, 8, 12, 2]
s_p_q = [8, 4, 8, 16, 8, 4]
totalBars = 0
for idx, x in enumerate(bars):
    totalBars = totalBars + x
    steps.append(totalBars * s_p_q[idx] * 4)
    print(totalBars)
conditions = [False, False, True, True, True, True]
injections = [False, True, False, True, False, False]
temperatures = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0]
primers = ["wativ_arp_187bpm_14bar.mid", "primer.mid", "primer1.mid",
           "primer2.mid", "primer3.mid", "primer4.mid"]
B.3 Etude III. 1712 Steam Engine (for human performer)

bundles = ("neutral_onF.mag", "booker.mag", "wativ.mag")
filenames = ["primer.mid", "primer1.mid",
"III.human_musicPrimer_wativ_slow18bars3.mid"]
steps = []
bars = [20, 8, 8]
s_p_q = [1, 2, 2]
totalBars = 0
for idx, x in enumerate(bars):
    totalBars = totalBars + x
    steps.append(totalBars * s_p_q[idx] * 4)
    print(totalBars)
conditions = [True, True, True]
injections = [False, True, False]
temperatures = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0]
primers = ["wativ_slow18bars.mid", "primer.mid", "primer1.mid"]
bundles = ("wativ.mag", "angry.mag", "booker.mag", "wativ.mag", "duo.mag", 
"wativ.mag", "duo.mag")
filenames = ["primer.mid", "primer1.mid", "primer2.mid", "primer3.mid", 
"primer4.mid", "primer5.mid", 
"IV.machine_speechPrimer_sam_angry_7bar_200bpm2.mid"]
steps = []
bars = [16, 4, 4, 4, 4, 4, 4]
s_p_q = [4, 32, 1, 16, 2, 32, 1]
totalBars = 0
for idx, x in enumerate(bars):
    totalBars = totalBars + x
    steps.append(totalBars * s_p_q[idx] * 4)
    print(idx)
conditions = [False, False, False, False, False, False, False]
injections = [True, False, True, False, True, False, False]
temperatures = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0]
primers = ["sam_angry_7bar_200bpm.mid", "primer.mid", "primer1.mid", 
"primer2.mid", "primer3.mid", "primer4.mid", "primer5.mid"]
B.5 Etude V. 1844 Telegraph (for human performer)

bundles = ("wativ.mag", "neutral_OnF.mag", "duo.mag", "wativ.mag",
    "monk_60sec.mag", "amused_OnF.mag", "monk_60sec.mag")
filenames = ["primer.mid", "primer1.mid", "primer2.mid", "primer3.mid",
    "primer4.mid", "primer5.mid", "V.human_music_wativ_drone_12bar7.mid"]
steps = []
bars = [16, 2, 8, 12, 2, 4, 4]
s_p_q = [4, 1, 4, 8, 2, 4, 1]
totalBars = 0
for idx, x in enumerate(bars):
    totalBars = totalBars + x
    steps.append(totalBars * s_p_q[idx] * 4)
    print(idx)
conditions = [False, False, False, True, False, False, False]
injections = [False, False, True, True, True, True, True]
temperatures = [1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0]
primers = ["wativ_drone_12bar.mid", "primer.mid", "primer1.mid",
    "primer2.mid", "primer3.mid", "primer4.mid", "primer5.mid"]
B.6 Etude VI. 2040 Full Immersion (for machine performer)

bundles = ("amused.mag", "wativ.mag", "amused.mag", "wativ.mag"
        "angry.mag", "duo.mag", "monk_60sec", "neutral.mag")
filenames = ["primer.mid", "primer1.mid", "primer2.mid", "primer3.mid",
        "primer4.mid", "primer5.mid", "primer6.mid"
        "VII.machine_speech_bea_amused_6bars_120.mid"]
steps = []
bars = [24, 8, 8, 8, 8]
s_p_q = [4, 16, 32, 1, 2]
totalBars = 0
for idx, x in enumerate(bars):
    totalBars = totalBars + x
    steps.append(totalBars * s_p_q[idx] * 4)
    print(idx)
conditions = [False, False, False, False, False, False, False, False]
injections = [False, True, True, True, True, False, True, True]
temperatures = [1.0, 1.0, 1.0, 1.0, 1.0]
primers = ["bea_amused_6bars_120.mid", "primer.mid", "primer1.mid",
        "primer2.mid", "primer3.mid", "primer4.mid", "primer5.mid",
        "primer6.mid"]
B.7 Etude VII. 2045 The Singularity (for human and machine performer)

bundles = ("angry.mag", "angry.mag", "angry.mag",
"wativ.mag", "wativ.mag", "neutral.mag", "wativ.mag",
"booker.mag", "duo.mag", "wativ.mag")
filenames = ["primer.mid", "primer1.mid", "primer2.mid",
"primer3.mid", "primer4.mid", "primer5.mid", "primer6.mid",
"primer7.mid", "primer8.mid", "VIII.machine_music_wativ_img_2440_9.mid"]
steps = []
bars = [7, 5, 7, 1, 8, 4, 1, 4, 2, 3]
s_p_q = [4, 12, 32, 32, 32, 6, 6, 2, 1, 4]
totalBars = 0
for idx, x in enumerate(bars):
    totalBars = totalBars + x
    steps.append(totalBars * s_p_q[idx] * 4)
    print(totalBars)
conditions = [False, False, False, True, True, False, True, False, True,
              False, True]
injections = [True, True, True, True, False, True, True, True, False,
              True, False]
temperatures = [1.0, 1.0, 1.0, 1.0, 1.0, 1.4, 1.0, 1.0, 1.0, 1.0]
primers = ["wativ_img_2440.mid", "primer.mid", "primer1.mid",
"primer2.mid", "primer3.mid", "primer4.mid",
"primer5.mid", "primer6.mid", "primer7.mid", "primer8.mid"]
References


Vita

William A Thompson IV or ”WATIV” is a composer, pianist, electronic musician and educator. Thompson studied jazz piano at the University of New Orleans where he earned an undergraduate and Masters degree in jazz studies. Currently Thompson is a PhD candidate at Louisiana State University’s Experimental Music and Digital Media program where his primary focus has been investigating and creating compositions and performances that deal with the intersection of music and human speech. Other notable projects while attending LSU include the creation of a acoustic piano augmentation called the Infastain Piano and his work in the compositional studio of Dr Mara Gibson.

His unique music has attracted attention NPR’s ”All Things Considered” and is the subject of a BBC radio 4 documentary. Thompson’s art has been most profoundly impacted by his one year tour of duty spent in Baghdad during the Iraq War in 2004 as a Counterintelligence Agent. The album release, Baghdad Music Journal was the result of that deployment and is the first album ever to be released from a combat theater.

Additionally Thompson works as a professional musician in New Orleans, La where he leads bands such as WATIV, Trapper Keaper, The Red Organ Trio, and The Betty Shirley band. He is also very involved in education and teaches “Composition for Digital Media” at Tulane University and “Sound Design” at Liberty Magnet High School. At the time of this paper’s completion, Thompson has accepted a position as Visiting Assistant Professor of Sound to begin in in fall 2022 at the University of Southern Mississippi.