Interdisciplinary Research Methods
EST 441

Project Report:

Neural-Knob:

Machine Learning in Music Technologies

Spring 2019

Professor:

Kevin Moriarty

Author:

Sudhin Domala #110475495

Table of Contents

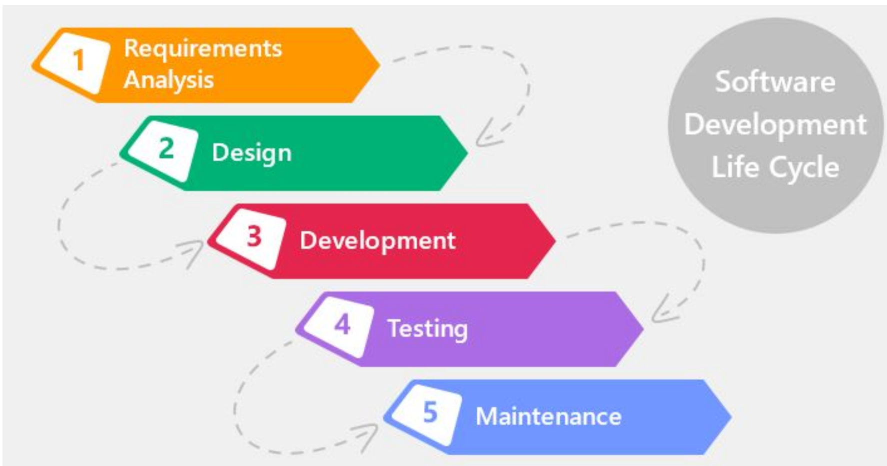
I. Introduction

A. Background

This project emulates the extent of computational creativity that one technical concept can have on the entertainment industry and specifically in the music technology sector. The music industry embodies many different individuals of skillsets in its workplace from performers, technicians, to distributors. Technology now is an integrated framework that connects these individuals to the market and to each other via communication, sound engineering, and electronic/digital instruments. Thus the application of an upcoming but latent form of computing or “Artifical Intelligence” will propel the application of song creation/selection to the next level in the terms of the consumer/artist specifications.

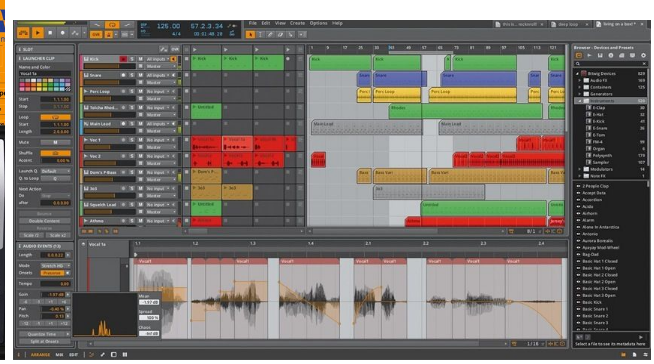
Prior to discussing the application, it is important to distinguish Artificial Intelligence is defined to be the cultivation of Big Data, Data Analytics, and Machine Learning. Although Artificial Intelligence could be said to be a better term for all the implications this type of technology has on the music market, the term, “Machine Learning”, is more appropriate to emphasize the possible learning capabilities to push this genre of entertainment. To clarify, any instances of Artifical Intelligence mentioned in this paper indicates the general picture of these computations that not only involve machine learning. However, Learning is key for this vision to ever be able to perform to a universal scale and that's where ML comes in.

The primary genre for the incorporation of such technology is electronic dance music due to its various computer hardware (synthesizers, portable audio interfaces) and software (Digital Audio Workstation and Virtual Instruments plugins). Given the nature of AI, the project will seek to develop a product around software deployment on consumer machines and follow a general software development life cycle.



B. Mission

This project’s mission is to facilitate EDM artists through the use of AI as a tool to augment human music generation. **NeuralKnob**, The core software product, will be designed for correlating song proportions to music style, interaction, and optimization under user discretion.



C. Benefits Realization / Quality Requirements

Many individuals have many troubles to understand (let alone incorporate) new technologies for their creative processes. Benefits of the software product can be not realized and quality of product use can be undermined given a not tech-savvy audience. Thus to solve this issue, management (for the development Life Cycle) and consulting will be an essential part of this software deployment to provide the best customer service. By educating the staff to provide commendable and professional consulting services, we can partake to improve a wide range of music industry disciplines down in the road. In-depth understanding of Music Theory and user tendencies will improve the chance for benefits realization. Music Format (MIDI vs audio), Database Design, and Data Analytics are also some parameters to where to check for the breadth and depth of product usage. Updates and quality checks will be part of the Prototype/version Testing and Maintenance in the Software Development Life Cycle.

II. Body

A. Technologies and Innovative Procedures

1) Background Research

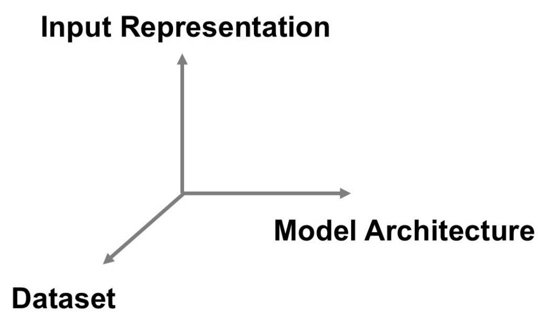
From remixes to original creations, A.I. can have the ability to merge the nuances of the genre and will be crucial to wider applications. In essence, computational creativity is one of the main subtopics that will be attempted to be defined and structured for such desired results. This creativity is an interactive medium towards artists to be able to explore the capabilities of AI systems through their customized algorithms. All in All, artists could greatly benefit from such technology if they **“think about AI as a tool for the augmentation of human thought and creation and make every effort not to turn the reigns of creativity or ethics over to the machines”**. Artificial Intelligence or the progress of Machine Learning has the requirement to be taught proper ethics and company practices in order to be reliable or even viable in such a respected market. Maintaining a sense of civility, creativity, and equity in the AI system will greatly justify this utilization. This can be done by following The Waterfall Model for the Software Development Life Cycle Stages.

Thus, this section explores the innovative procedures to be made in the Requirements Stage and Software Design Phase. Taking note on the technical limitations that may arise in the client’s side and following ready-to-follow specifications that meet the needs of clients. Being documented in a problem vision document with a “business vs success” criteria in mind. All in all, The primary idea of this future application needs to be set in stone with requirements gathering to progress in the development cycle without being sidetracked. Questionnaires, interviews or just basic access to current technology preferences are some examples for such client-based information gathering.

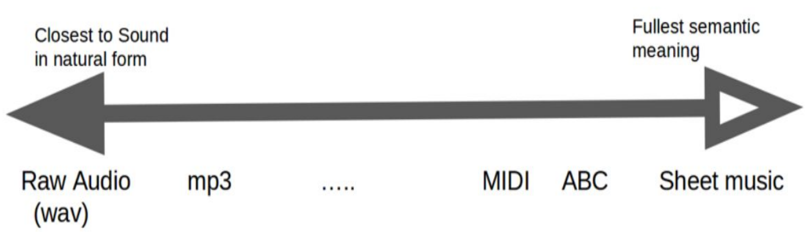
2) Music Similarity Estimation

There are many types of EDM music from House to Dubstep. Give such a variety and what a specific artist prefers to stick with, the first obstacle for our software product is to tackle what genre manner (and the extent of it being used) the artist wants to incorporate his/her music in. This hurdle is hard to cross but crucial to base any Artifical Intelligence concepts like Machine Learning or Data

Analytics (ex: Visualization) further in the development cycle. These processes should be hidden and automatically done with minimal user interference.



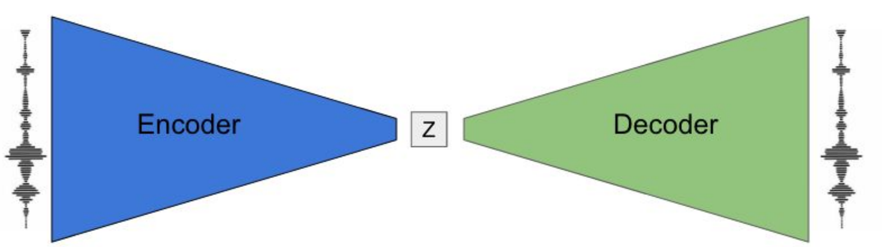
In the Design Phase, the important factors are best presented in an (x,y,z) graph with Input Representation, Model Architecture, and Dataset respectively. In terms of input, music can either be a melody (single instrument) or polyphony (multiple instruments) and what degrees are these in accompaniment (being complementary or supplementary to another musical part). In a metric manner, there are various forms of measurement that exist in music technologies depending on the sound from the closest to the natural form (raw audio like wav and mp3) to the fullest semantic meaning (MIDI, ABC, sheet music, etc.). These attributes are all part of the first stage of Machine Learning: Data Pre-Processing.



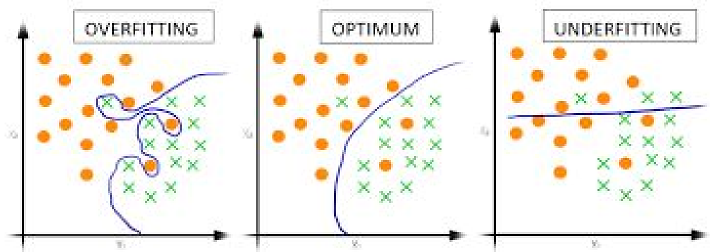
“Musical Instrument Digital Interface” (MIDI) is the biggest and easiest factor to quantify and translate for our technology purposes in the EDM sphere (at least). This format can be said to a uniform “sheet music” for various instruments or plugins at the client's disposal. This format can be codable in the functional sense where u can set actions like setting tempos/notes in event-by-event iterations. Other representations can be in the timbre similarity as in the tempo BPM (beats per minute) and what note (pitch or frequency) is being played. This will help for Visualizations and statistics to look back on later down the road. These can be represented in "decision templates" (where what instruments are used and to what extent in the piece). In this graphical case, the templates are just another name for datasets that the model architecture takes.

For instance, a distribution method to distinguish between the "signature" of two songs "formed by the clustering of Mel-frequency Cepstral Coefficients (MFCCs), calculated for 30-millisecond frames of the audio signal, using the K-means algorithm." This is a classification algorithm that will distinguish the genre and artist identification to form playlist generation to be pushed for ML evaluation.

While all these components will be graphed under one representation of a specific instance in the z-axis as a dataset, The Model Architecture will be where the format of the Machine Learning will take place. It can be in Sequence-by- Sequence basis where there is an encoder, a decoder, and their own relative lengths to formulate an experimental setup for test output (in units of datasets).



Following the output comes the second and third stage of Machine Learning: Testing. This is where you fetch the sequence (input dataset) to predict the steps that follow. Every time there is a prediction, the weights (significance of each step) changes to “Update” and “Repeat” the observed data to output the desired result (be it a playlist of likable songs/song bites or genre-oriented melody that is most optimal to follow with). Thus concludes the Model Architecture part of estimation for what music is similar to one another.



Over-fitting is the biggest challenge when it comes to this estimation. It is essential to avoid reduced accuracy in where the threshold exists of what music pieces are similar to one another while what aren’t. This can’t be overlooked thus giving outlier suggestions to the client that will evidently lead to a lack of customer satisfaction. Every ML algorithm and data discrimination method have optimal results for different parameters given the extent of music format that's out there. Our consulting department and primarily the SDLC Testing phase will be on the task to evaluate which version of the software best fits the user’s intentions in finding new music pieces or exploring the nuances of his/her own.

Keep in mind, this technology primarily fits in composing DJ sets and artistic experimentation of an existing reservoir of songs the artist wants to explore/build upon. In the musical sense, the composition is the way in which you exhibit the music and is not the interactive creation that the next section delves into. Remixes within the genre or under the guise of a different one can push the artistic expression that a song can exhibit. Machine Learning's classification algorithms and visual graphs that are derived from this technology will serve this end what our software has to offer.

3) Music-Machine Interactivity

This technological procedure will the secondary form of innovation to facilitate our users' experience learning and adopting our product in their creative workflow. As mentioned in the need of consulting, a significant part (if not most) of our users will need the framework of how our products generate music parts to overall manipulate the inputs to configure their desired inputs. Explaining the machine learning process in one opening tutorial will be very daunting for the user thus working on a prototype software toolkit about our processes will solve this issue. In essence, this toolkit will "provide the musician the ability to iterate through modification and testing stages to improve the design."

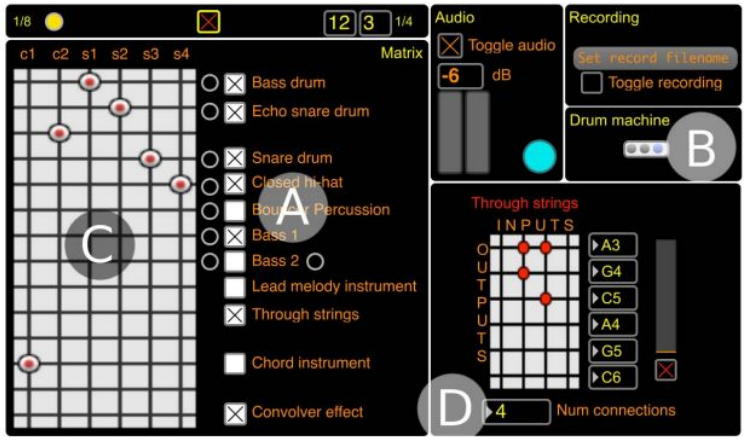


Figure 1. User interface of a software music system which could be controlled by a musician and an IMA in tandem. The labels A-D are referred to in the text.

Over time, users can make more settings and alignments to even design their own artificially intelligent agents. There will be high-level musical decisionmaking about the sequencing and juxtaposition of preprepared segments of musical material such as sound samples, MIDI sequences, or low-level generative processes (for example, a drum-beat generator). This design workflow will follow the same concepts of encoding and decoding the music segments at hand. The training data set will comprise of prepared segments and recorded performances. The toolkit will analyze this set and extract whatever patterns and behavior it finds.

While this decision-making will first be made by ML algorithms, the musician plays a critical role in guiding these algorithms to be the environment for the effective model he/she foresees. Examples of these algorithms can be seen under Google Magenta's Neural Synthesizer ([NSynth]) Instruments and [Performance RNN], "an LSTM-based recurrent neural network designed to model polyphonic music with expressive timing and dynamics."

This design process will essentially in three parts:

1. Choosing the parameters of algorithms that learn how the musical elements are sequenced in time;

2. Choosing groups of musical elements that are interdependent so that rules may be discovered describing the dependencies

3. Identifying particular features of the example performances that should be reproduced by the agent, and configuring the software so that these features are recognized.

Overall, the Interactive music Agent (IMA) may then be evaluated through real-time interaction. If the behavior model isn't up to par. Then the musician can either add more examples to better illustrate the desired behavior, re-configure the machine learning algorithms or even manually alter the behavior model. The following diagram highlights these options in the artist's creative workflow.

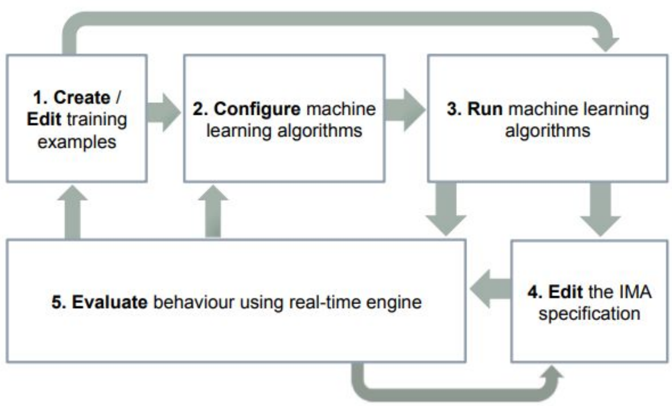


Figure 2. The interactive machine learning workflow supported by the toolkit.

All in all, these toolkit features help make the user interface more accessible and familiar to be the context of a real working scenario for electronic musicians. Leaning in making the interface supported in Ableton Live will be the relatable platform for layering/sequencing that modern EDM artists have to be inept at.

2. Business Plan Prospects

Overall, our business will consist of software engineers, consultants, digital marketing managers and should be around 15 people. However, given the nature of this technology, prototype testing will take the majority of the first year at the very least. There will be efforts to seek funding via sponsors, angel investors or overall online donations to demonstrate the ingenuity and power of our product for EDM artists that are popular or upcoming.

1) Project Risks

AI risks are broad in the sense of how it will be to pitch our software to be reliable and effective for music creation/experimentation.In the development, we might overestimate the power of the AI techniques at our disposal. This can lead to not have refined processes in the terms of our users and to ultimately have NeuralKnob to be not quality-assuring. Algorithmic bias and Programmatic errors give discrimination and confusion respectively between the relationship of the designers and users. Having a online service means there exists is a degree of needed cybersecurity. This touches on copyright infringement and loss of personal data that will heavily deter any user to use this platform. This can spiral to having legal risks and liabilities in our access to song segments and private artistic practices that a artist will not be found in sharing. These risks succumb to prioritizing our reputation as the proper bridge of higher level computing and music creation.

2) Copyright Infringement

Copyright Infringement is a huge concern when dealing with artist input versus their correlation with other music domains. Making remixes or different covers will need proper labeling and filtering to whether it is up for publication and usage. Having proper filtration and boundaries on our software will be essential for computation scalability. Having this being a priority in the Design Phase can lead to better database automation and public/legal approval. Music Subjectivity is what will tip the scales of whether the outputs of this software in unison of artist creation is viable for marketable usage. Finally, this extends making sure if our product will be open source or a legible revenue opportunity when pairing with the results to the present copyright laws.

III. Conclusion

Modern technologies like IBM Watson and Google Magenta have established indicators of “robust understanding necessary to create A.I. music curators and even A.I. music critics.” Illustrating the idea of computers being able to break/analyze music down to its components is a principle already applied in Pandora and Spotify. The future of this technology resides in the pursuit that A.I curators no longer need to be exclude in the music creation process of coming musicians exploring their voice and genre.

Having NeuralKnob as an extension or a plug-in for modern music software is a lot more implementable for the current music industry. Overall, the logistics of this software development were outlined in detail and overall promising to be out of research settings and in the industry as a marketable tool. This technology can be an iterative process to pave a way for humans and computers to be more creative is the proper gesture to conclude this project. Thus, this technology is nothing different from being a different instrument to stand out from the crowd (or toward the consumer from other use cases). Machine Learning "will push how fast learning advances in terms of generative models, but artists are the owns that will work faster to push the boundaries of what's possible." Like the surge of EDM music in this decade, true beauty or ingenuity can found in Machine Learning and a person's musical ability in collaboration.

III. References

https://dl-acm-org.proxy.library.stonybrook.edu/citation.cfm?id=1178737

https://dl-acm-org.proxy.library.stonybrook.edu/citation.cfm?id=2071567

https://dl-acm-org.proxy.library.stonybrook.edu/citation.cfm?id=2481211

https://dl-acm-org.proxy.library.stonybrook.edu/citation.cfm?id=1878016

https://dl-acm-org.proxy.library.stonybrook.edu/citation.cfm?id=2973795

https://magenta.tensorflow.org/

http://www.sciencemag.org/news/2017/08/how-google-making-music-artificial-intelligence

https://view.officeapps.live.com/op/view.aspx?src=https://cdn.oreillystatic.com/en/assets/1/event/280/Computational%20creativity_%20Making%20music%20with%20AI%20technologies%20_sponsored%20by%20Microsoft_%20Presentation.pptx