A Probabilistic Jazz Automaton for Sharp11 – Proposal

For CPSC 290, I would like to design a non-deterministic finite automaton to analyze jazz chord changes, and then train the automaton on a dataset to produce probabilities on the transitions that would allow new music to be generated stochastically. I will be formally working with Scott Petersen of the computer science department and Brian Kane of the music department, and I will also be consulting with Ian Quinn of the music department who is on sabbatical next semester. The deliverables for this project will be a program which can generate a sequence of jazz chord changes and a write-up explaining how the program works, what decisions were made in constructing the automaton, and why those decisions were made.

I will implement this program on top of Sharp11 (https://github.com/jsrmath/sharp11), a JavaScript music theorization and improvisation engine that I developed and have been working on for several years. Sharp11 is an open source project that I intend to keep expanding on to create an open-source computer music ecosystem with a jazz focus. I see this project as a part of that ecosystem. Building on Sharp11 will also allow me to avoid reinventing the wheel with respect to certain basic music theoretical operations that I will need to perform. That said, I will be very clear about what code was written specifically for this project.

I foresee the process for this project as follows: design a theoretical model for the automaton, implement it, train it on real song data, examine the output, tweak the model, and iterate. The theoretical model will, at least initially, be based on a music theoretical framework called functional-bass analysis, which is being developed by Professor Quinn and is used by Professor Kane in a jazz context in MUSI 305 Jazz Harmony, which I completed last semester. I seek to formalize a variant of this framework computationally and use it to construct an initial automaton.
After designing the initial automaton, I will construct a more thorough automaton by applying operations to the initial automaton. I will define these operations by analyzing and formalizing jazz harmonization techniques. The main text I will get these techniques from is *The Jazz Theory Book* (Mark Levine, 1995). I will also retain names and descriptions of these operations, so that running a song through the automaton can provide information about what harmonization techniques the composer used and where. In my write-up, I will describe rigorously how to apply these operations to yield the final automaton, and then implement them as such in the program.

The data that will be used to train the automaton for the generative portion of the project will come from The Ohio State University’s iRb Corpus (Yuri Broze and Daniel Shanahan, 2012), which contains 1,186 songs in a jazz format. The jazz format is standardized and contains all the information necessary for the analysis that I’m doing. It is available free online at [http://musiccog.ohio-state.edu/home/index.php/iRb_Jazz_Corpus](http://musiccog.ohio-state.edu/home/index.php/iRb_Jazz_Corpus). Completing this project also entails developing tools for analyzing and querying data from this corpus, which I plan to document and hope will prove useful for other projects.

As I mentioned, I will be working with three professors. The roles they wind up taking might change depending on how the project progresses, but my initial idea is as follows. Professor Petersen will advise me in an official capacity and will help me with the more computational parts of the project, such as constructing and training probabilistic automata. Professor Kane will be my primary reference for jazz theory and practice. There is not a definitive guide to what is and isn’t jazz, so I will be relying on his expertise in making decisions about how to construct my model and analyzing the quality of its output. Professor Quinn, who will be returning to campus in March, will help me in developing and documenting the
underlying music theoretical model that the computational model will be based on. He will also help me analyze the results of my model in a rigorous way.

I also have a number of stretch goals for this project. If I complete the project quickly enough, I’ve selected two that I’d like to focus on for this semester. Either way, I am planning to continue this project in some form in the subsequent semester for CPSC 490.

1. Harmonic rhythm and song structure. At the moment, I only intend to generate series of viable jazz chord progressions, but not take into account how long each chord change lasts or where it falls in the structure of the song. In the future, I would like to be able to generate the chordal part of jazz lead sheets, that is, a list of chords and durations with some sort of canonical song structure.

2. Modulation. I am currently only designing my model to work in a single key at a time. That said, many jazz tunes change keys at some point, and if and where a tune modulates is often subject to interpretation. Currently, a song that modulates will either be rejected by the automaton or parsed in some sort of nonsense way. When training the automaton, these songs will effectively be noise. In the future, I would like to develop a way for the model to take into account modulation as a possibility, both on the analytical and generative ends.

As part of the Sharp11 ecosystem, the project will be written in JavaScript and hosted on my personal GitHub. I also hope to eventually yield a publishable paper from this project.