Final Project: Musical Memory

Jeff Kaufman

May 12, 2008

Abstract

This paper presents a machine learning system for notes, capable of learning some aspects of tunes. Input is in the form of notes played on a penny whistle, output is a prediction of the next note. In testing the system performed far above chance, but there are some important forms of generalization that it cannot perform.

1 Introduction

In this paper we present a learning system for notes, along with software and hardware for using it on live music. The overall project goal, of which this is only a portion, is to create a robot that can learn to play penny whistle. [1] This robot would, starting with completely unanalyzed sensors and effectors in the manner of SODA [2], learn in an unsupervised manner to play both along with other people and alone. An overview of the system as implemented so far is shown in Figure 1.

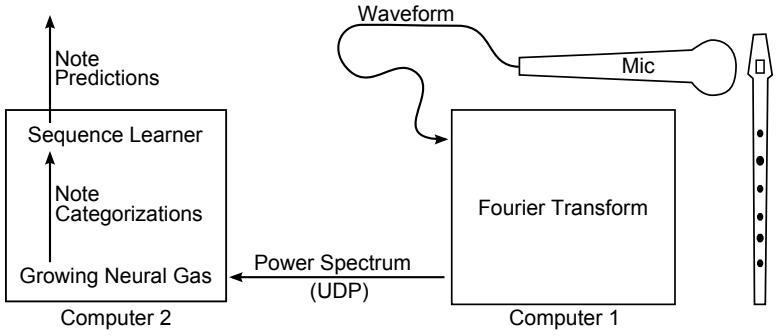

Figure 1: System Overview

[1] A penny whistle is a six-hole fipple flute, capable of playing a D-major scale (with the addition of C natural) over two octaves. We chose it for this project because it produces clear notes that respond well to computer analysis and because it has simple fingering.

For the class midterm project we implemented the first two stages of this process. The first stage is a low level stage with no learning component. First a computer samples the analog output of a microphone at 44100 Hz using 16 bit samples. Then, every 1102 samples or 40 times a second, it computes a power spectrum with a Fourier transform on the sampled waveform. The spectrum consists of 501 frequency 'buckets', each representing the strength of a 40 Hz wide slice of the sound spectrum.

The second stage is an implementation of a Growing Neural Gas (GNG) learning system as in [1] that creates categories corresponding to notes. It receives a power spectrum over UDP 40 times a second, updating and adding internal 'prototype' nodes. Over time it learns to distinguish common frequency combinations (notes), which can then be given as input to the next stage. While a biologically realistic system would likely have the GNG continuing to form and update categories, this is experimentally inconvenient, as it can make later experiments less consistent between runs. Instead, we trained the GNG on twenty minutes of human penny whistle playing and then 'froze' it. After initial training it functioned only as a categorization system, producing symbols 40 times a second indicating how it was categorizing the current sound.

The third stage and focus of this paper is fundamentally a prediction system. The goal is to predict what note will be detected next given the sequences of notes that have been heard in the past and the recent history. A biological approach suggests 'short-term' and 'long-term' memory components. Short term memory is implemented as a simple list of the previous notes heard, going back fifty notes. Long term memory is implemented as a kind of trie. An example prediction trie is shown in Figure 2.

The trie begins with only a root node, indicating an empty memory. Initial predictions are 'null' and so always incorrect. Upon experiencing a sequence, if the final note (t_1 or p) is predicted correctly by the trie given its history (t_0 through t_{-n})[2] The final note of a sequence is then the trie is left unchanged. Otherwise, a new node is added at $t_{-(n+1)}$ as a child of the t_{-n} node the search ended at, predicting p.

[2]Note that here we are indexing in the opposite manner customary. If the sequence 'd a b c' is detected, for example, then t_1/p is 'c', t_0 is 'b' and so on.

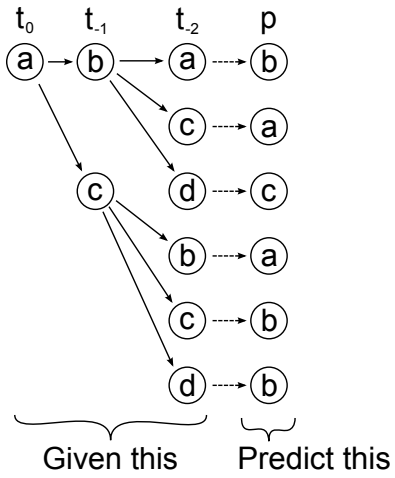

Figure 2: An example prediction trie

2 Experiment

In order to evaluate how well the system can learn sequences, we tested it on six tunes. Each tune was played six to ten times on a penny whistle. After hearing every transition between notes, the learning agent predicted what the next transition would be. We can evaluate accuracy by looking at the fraction of transitions predicted correctly.

The six tunes do not give the appearance of uniform complexity. Table 1 shows the tunes ordered by the average note duration.

Tune Name	Tune Length	Number of Notes	Average Note Length
Row Row Row Your Boat (Em)	54s	170	0.312s
Row Row Row Your Boat (Em) (complex)	44s	264	0.166s
Jamie Allen	105s	788	0.133s
Canadian Barn Dance	103s	1073	0.096s
Evit Gabriel (A part)	32s	544	0.059s
Swallowtail Jig	67s	1195	0.056s

Table 1: Complexity of Tunes

3 Results

The learning system was able to learn all of the tunes to some extent. Table 2 shows how the system did on each of the tunes overall starting from a blank slate, while Figure 3 shows a graph

of accuracy over time for one tune.

Tune Name	Percent Incorrect
Row Row Row Your Boat (Em)	44.1
Row Row Row Your Boat (Em) (complex)	58.3
Jamie Allen	42.3
Canadian Barn Dance	57.4
Evit Gabriel (A part)	61.3
Swallowtail Jig	57.7

Table 2: Percent of Note Transitions Predicted Incorrectly

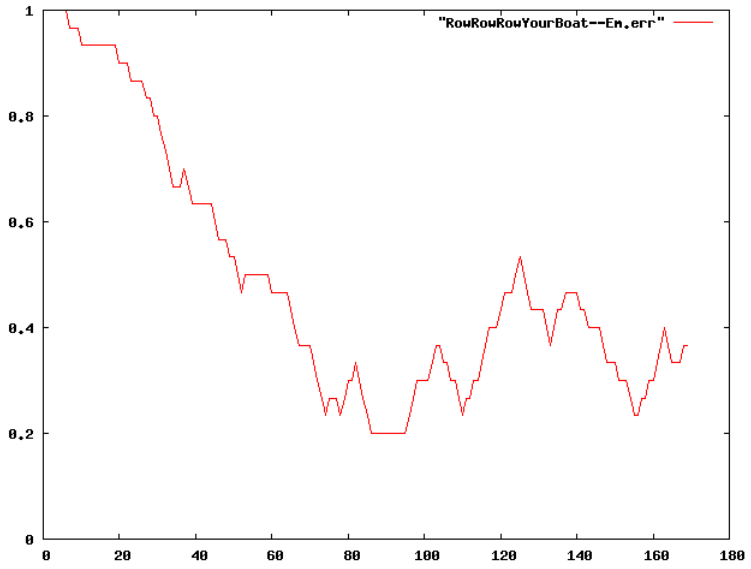

Figure 3: Percent of transitions in "Row row row your boat (Em)" incorrectly predicted, 30 transition moving average

In looking at Table 2 it is important to note that on the first few transitions error is going to be 100% because the system starts with a blank slate. A "percent incorrect" score of zero then would not make sense, as it would indicate a psychic learning mechanism. There is certainly room for improvement in this system, but these rates are not as bad as they might appear.

One curious feature of Figure 3, and in fact of the prediction patterns for all the tunes, is that while prediction error goes down heavily at first it often reverses its path later. We are not completely sure why this happens, but suspect it is due to our system not taking into account errors in the input stream. A person playing a penny whistle may occasionally play a wrong note, and the prediction trie is not very robust in the face of error. Lookups in the trie do use preferentially more recent samples and so errors a few notes back may not cause too many errors. The transition

immediately after an incorrect note, however, will nearly always be mispredicted. Use of a different learning mechanism that copes with a noisy input stream could help with this and would be an interesting direction for future work.

Other directions of future work include making the learning system aware of timing and duration, modifying the low level spectrum producing algorithm to use wavelets for better accuracy, finishing construction of the hardware for robotic penny whistle playing, and implementation of an IAC-style learning mechanism for supervision and control of the hardware.

4 Conclusion

With this project we have shown that one can make a system capable of learning to predict note categories starting with completely unknown sensors. This system would be an essential component of a full robotic penny whistle learning system.

References

[1] Bernd Fritzke. A growing neural gas network learns topologies. In G. Tesauro, D. S. Touretzky, and T. K. Leen, editors, *Advances in Neural Information Processing Systems 7*, pages 625–632. MIT Press, Cambridge MA, 1995.

[2] Jefferson Provost, Benjamin J. Kuipers, and Risto Miikkulainen. Developing navigation behavior through self-organizing distinctive state abstraction. *Connection Science*, 18(2), 2006.