Computing BEng

Individual Project

Rubato: A Musicality Tutor using Adaptive Learning Techniques

Conrad Godfrey

Supervisor:
Iain Phillips

February 21, 2014

Contents

Chapter 1

Introduction

In this paper I propose an application to harness the power of the modern web browser to deliver engaging musicality tutoring based on adaptive learning principles. The application will comprise of various exercises designed to challenge and improve a user's musicality, and keep track of their progress as well as adapt to their strengths and weaknesses in different areas.

As usage of the world wide web has become ubiquitous among citizens of the developed world, web based teaching has grown rapidly to try and modernize learning methods to fit into the 21st century lifestyle. Increasing numbers of companies like Coursera,[1] Codeacademy,[2] Duolingo[3] are providing easily accessible education for anyone with an internet connection and a desire to learn. Codeacademy and Duolingo in particular are notable for their use of rich, highly interactive learning tools which - through requiring the user to have an input into the educational process - establish a feedback system with powerful results.[4] These companies offer a variety of different programmes the user can study, and each of these is taught through a series of exercises, the results of which are used to track the users progress through the course, and provide statistics on how much they have learned. Duolingo, a website that teaches foreign languages, takes it a step further by introducing adaptive learning methods to recognise the user's proficiency in various areas, information it can then use to tailor-make lessons to fit the user's needs. Duolingo is arguably the best best-known example of an adaptive learning system put into practice. It will therefore often be used for comparison throughout this report, as it has also provided a lot of inspiration for my thinking about the direction my project will take.

The web is a great place to teach music theory, and train the musical ear, a pair of skills we shall refer to under the umbrella term of "musicality" from hereon in. The rich media options possible on modern day web browsers mean that the input and output of music to a web browser is not only easy to implement, but easy to make user friendly. However, while there are many "music theory tutors" and "ear trainers" available, there are no existing adaptive learning solutions. Such a solution would hopefully be invaluable to potential learners as the current best solutions still have no idea who you are and what you've achieved after you leave the page.

In order to teach adaptively we must analyse information about a user's performance, and then change how we teach accordingly. There are various ways of measuring musicality, which will be discussed in more detail later, and we will rely on these metrics to adapt the learning experience to fit the user.

1.1 Objectives

I aim to build a program that can successfully teach musicality. To ensure I achieve this goal, the following criteria must be met:

- **Adaptive** - The product must analyse the user's progress and their strengths, and adopt the content of their learning experience accordingly. This will require a user account system, as well as intelligent handling of the exercise data they generate.
- **Intuitive** - The product must be simple to use, and the exercises given to the user must be simple to understand, even for a beginner.
- **Engaging** - The user must want to learn and continue learning. In a study of Duolingo's effectiveness, 93.8% of participants intended to continue using the website after the study had finished.[5] I would like to aim for at least 75%.
- **Pitch Recognition** - Another way I'm aiming push the boundaries of musicality teaching is through introducing exercises that involve pitch data to be submitted by the user, and then grade them based on how accurate they are.

Chapter 2

Background

2.1 Adaptive learning

Web based learning has been a topic of interest almost since the birth of the web,[6] and the introduction of the HTML5 standard has opened up even more opportunities for computers to play a role in the education process.[5] 3 notable examples of web based learning systems (WLS) are:

- Coursera, a site that provides lecture material for various university level courses.
- Codeacademy, a site designed to help people learn coding by getting them to undertake interactive browser based programming lessons.
- and Duolingo, a site designed to help the user learn foreign languages through interactive exercises that tries to tailor-make each exercise to the users needs by analysing their strengths/weaknesses.

The end aim of these sites is the same as all WLS's: to educate the user, but there are key differences between them which we can use to categorise them further into a hierarchy.

- Coursera is a *static* WLS, it allows a one way interaction whereby the user can view/download learning material. While this is surely useful, it is really the base level of what a WLS can do.
- Codeacademy is a *dynamic* WLS, it allows a two way interaction that introduces feedback for the user, enhancing the learning experience beyond a static WLS.
- Duolingo is an *adaptive* WLS, it includes all the features of a dynamic WLS, only it can treat every user differently by analysing their strengths and weaknesses.[7] This adaptive approach allows for superior teaching to static or dynamic WLS's, as if implemented correctly it will start to mimic the tailor-made learning experience one might receive from a real-world 'for-hire' tutor.

2.1.1 Adaptive learning in music

There are many ways that adaptive learning techniques can be applied in a musical context. As an example, imagine a simple exercise.

- The user is played a rhythmical phrase.
- They must then replicate the phrase by tapping it in on the space bar.
- The application then calculates how accurate the user's approximation of the rhythm is, and feeds this information back to the user

After repeating this exercise multiple times the application notices something: The user is always getting examples featuring multiple consecutive dotted quavers wrong [1], and determines that this particular rhythmical device is something the user is struggling with. It can then subtly adapt future exercises to incorporate this device prominently, in order to expose the user to it as much as possible, and hopefully cause them to improve their understanding of that specific rhythm, and rhythm in general.

2.2 Platform choice

As the modern internet browser has become more and more powerful, web apps have been able to reduce the previous speed disadvantages the faced when compared with their native equivalents. The web-platform is advantageous due to having a singular codebase, meaning that it can be used by anyone with a web browser, independent of their device, meaning it has the potential to reach more users. Another key advantage of building a web app is that as user testing is so key to evaluating the app's success, when the time comes, and I want to show it to users, I can easily point people to my website and people will know how to get to it. Distributing a mobile app to others for testing purposes is painful, and difficult to update once they have installed it, with a web app I can instantly change the build of my website, and my aunt who's testing it in Jamaica won't have to do anything more than refresh the page.

2.3 Existing Musicality Tutors

There are a wide variety of existing web-based applications that teach musicality, to varying degrees. There are programs to allow you to practice interval recognition,[8] identify a note in a chord, practice recognition of rhythms.[9] A lot of these programs are standalone, and focussed on one specific area. Musictheory.net[10] is a good example of a website that that goes beyond that, it provides exercises that test a wide variety of skills, trains your ear as well as providing music theory lessons. However Musictheory.net

[1] This is a type of rhythm that could prove difficult for the user as it can sound like the rhythm is going in and out of time due to it's naturally syncopated nature against a four four baseline

provides no adaption to the user, the furthest it goes is allowing you to specify what you want to be taught, but it is not intelligent enough to work it out itself.

2.4 Pitch Detection

Pitch detection is a well understood problem that has been researched for many years. The pitch of a note, as described above, is the human perception of how high or low it is. The pitch value of a sound signal is determined by the fundamental frequency, f_0 of the signal, and thus the problem of pitch detection of a signal is analogous to finding that signal's fundamental frequency. The mapping of frequency to pitch is determined by the 'tuning system' you use, and in Western music, equal temperament is the most commonly used. The way it works is as follows:

- An f_0 of 440hz corresponds to an A above middle C, otherwise known as A440
- Doubling the frequency causes an increase in pitch of an octave, so A880 is the next A above A440.
- Since a leap of 12 semitones is caused by a doubling of the f_0 , it follows that the frequency ratio between each semitone is $2^{\frac{1}{12}}$ or about 1.059.

So after we've calculated f_0 it is simple to calculate the pitch. If we assign an arbitrary numerical value to A440 such as 69 (as used in the MIDI specification[2]), then we can use the following equation to determine the pitch of a given frequency, f, as a value in pitch space, p, where semitones correspond to a gap of size 1.

$$p = 69 + 12 \times \log_2 \frac{f}{440Hz}$$

2.4.1 Fundamental Frequency Detection Methods

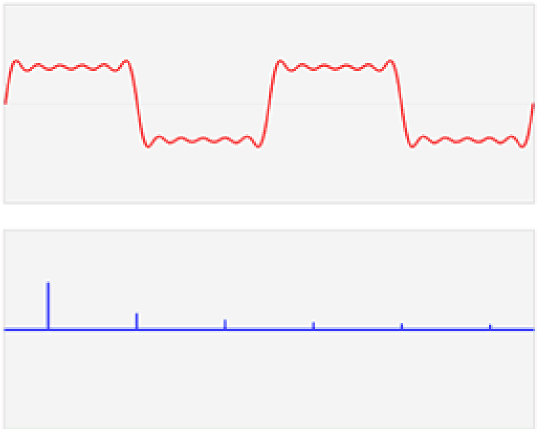
f_0 detection is a difficult process, and there is no 'best method' so to speak, each approach has its drawbacks and advantages, the normal tradeoff being that a method that is fast may not be reliable and vice versa. There are two approaches to f_0 detection, analysing the signal in the the time-domain, or the frequency domain. To analyse in the frequency domain requires the use of the Fourier Transform, and is therefore a slower process, but provides more accuracy. However, as we are only doing monophonic pitch detection, and responsiveness is key, then for our purposes, time-domain algorithms will be the most appropriate.

Autocorrelation

A standard method of time-domain f_0 detection is autocorrelation. It exploits the fact that a periodic or quasiperiodic waveform such as a sustained sung note will be self-

[2] MIDI is an industry standard for storing musical sequence data

Figure 2.1: Time domain (above) vs frequency domain representations of the same signal



similar by the definition of periodicity. If we compare a waveform with a copy of that waveform offset by the period of the waveform i.e. ${f_0}^{-1}$ then we should expect to see a strong correlation between them. So autocorrelation works by iterating over all the possible offset values, and determining which one has the best correlation. This correlation value for each different offset is described by the equation below, where x is the signal function, N is the window size of the waveform you are considering, and v is the offset value.

$$R_x(v) = \sum_{n=0}^{N-1-v} x[n]x[n+v]$$

Chapter 3

Evaluation

In order to evaluate the success of my program, I will need to analyse each of my objectives to see how far I've come in acheiving them.

3.1 Adaptivity

How well I have achieved adaptivity in my teaching will be a difficult thing to test quantitatively, as there is no existing way of measuring how adaptive a learning system is. One way to gain a greater understanding of how effective the adaptivity is would be to use the program and artificially fail parts of exercises to try and prompt the system to adapt to my behaviour, and see how it responds.

3.2 Intuitiveness

The intuitiveness of my program is something that can only be judged by other people using it. I intend to involve others in the design process from the start, by using techniques like hallway testing to quickly get feedback on how users approach my app, and what they would change about it. At the end of the project I will also hopefully get as many different people as I can to use it, and fill in a survey at the end of their usage period detailing their experience with the app.

3.3 User Engagement

This will be handled by users answering questions about whether they would continue using it, and how much they enjoyed the process. These questions will be found on the survey described above.

3.4 Teaching Profficiency

To measure the general success of my product as a teaching tool, I shouldn't actually have to do too much, as (if succesfully implemented), my app should be able to track the users progress, and store information about how much they've improved.

3.5 Pitch Detection

Pitch detectors are prone to error, so it is crucial to evaluate their performance thoroughly. They can be thrown off for many reasons, including varied vowel sound, a different person's voice, and background noise, so it will be essential to test the detector in different environments that the app may face. Determining automatically the success of pitch detection algorithms requires you to be able to measure pitch correctly in the first place, a somewhat circular reference. Therefore the only way to do it will be to manually try out different environments, and measure how successfully the app classifies pitch in each situation. I will also try and find multiple people with different voice types to help test the pitch detection to further try and ensure it is as 'context-free' as possible. This will be a fairly arduous task that will need to be ongoing throughout the development process.

Chapter 4

Plan

So far I have begun the design of the website, as well as started to implement the first pitch detection function. I am aware that I have a lot left to do, so it is crucial that I plan my time correctly

- 3rd of March - I will aim to have the basic skeleton of my website working including user log-in, and a rough pitch detection algorithm in place.
- 14th of March - Improve accuracy of pitch detection algorithm, and implement it into a "Note from Chord" exercise.
- Suspend progress until end of term on 28th due to exams!
- With the basic web interface designed, create multiple different exercises over Easter break until 26th of April and start getting users to test them. Start to experiment with creating adaptability in exercises to see how it can be done.
- 28th of April - REALITY CHECK, see where I've come, and where I still need to go. Re-evaluate rest of plan up until June accordingly.
- 1st of June Begin intensive user testing to gather information for my evaluation write up project simultaneously.
- 17th of June Final Report due.

4.1 Extensions

If I finish ahead of time, there are a couple of extensions that come to mind.

- Support for MIDI instruments. There are a number of existing instruments you can plug straight into a PC using the MIDI to USB interface, it would be interesting to see how I could incorporate that into my project.

- Segmented crowd-sourced music transcription. Allow users to to test their music ability by playing them a short clip of music and getting them to supply information they've perceived about the segment, chord data, time signature data, melody data etc. The idea is that if enough users completed segments of a particular song, the song's structure could be pieced together.

Bibliography

[1] "Coursera, take the world's best courses, online, for free.." `http://www.coursera.org/`.

[2] "Codeacademy, learn to code interactively, for free.." `http://www.codeacademy.com/`.

[3] "Duolingo, free language education for the world!." `http://www.duolingo.com/`.

[4] R. Vesselinov and J. Grego, "Duolingo effectiveness study," *City University of New York, USA*, 2012.

[5] L. S. Griffiths, R. Ogden, and R. Aspin, "A profile of the future: what could html 5 do for he by 2015?," *Research in Learning Technology*, vol. 20, 2012.

[6] S. Alexander, "Teaching and learning on the world wide web," in *Proceedings of AusWeb*, vol. 95, p. 93, 1995.

[7] "Duolingos data-driven approach to education." `http://blog.duolingo.com/post/41960192602/duolingos-data-driven-approach-to-education`. Accessed: 2014-02-20.

[8] "Interval ear trainer - recognise the interval between two notes.."

[9] "The rhythm trainer - rhythm recognition and representation.." `http://www.therhythmtrainer.com/`,.

[10] "musictheory.net, on online music theory tutor and ear trainer." `http://www.musictheory.net/`.