# Computer-executed Genre Classification of Music
# Computer Systems Lab 2009-2010

Alex Stabile

October 26, 2009

## Abstract

The aim of this project is to enable a computer to place given pieces of music into their appropriate genres. Genres are often generalities that do not necessarily fit a piece exactly, but this project could determine what music really is similar and should be grouped together. This could have applications in sorting large libraries of music or suggesting music to individuals based on what they like to listen to. Music in midi format will be used as input for the program. Python will be used to write classes that can read and store the information held by midi files in an organized manner. Midi files contain exact note and rhythm information, and therefore lend themselves to analysis by a computer, as analyzing compressed sound data is not required. This project will focus on musical parts for a solo instrument, so that notes and rhythm rather than instrumentation are used to determine genre.

**Keywords:** music, genre, naive, bayes

## 1 Introduction and Background

Current research often uses statistical models to determine how a given piece of music should be categorized. One approach has been to use Inter-Genre Similarity modeling [1]. This project attempted to categorize music by analyzing the timbral textures of a short sample of music. These textures differ due to differences in instrumentation and rhythm. Gaussian Mixture Models were used to create the statistical model, and IGS to cluster similar groups together. Their algorithms grouped a 0.5 to 30.0 second sample of music into one of nine genres. Longer samples yielded better results, up to 64 percent correct. Another approach attempted an automatic method of classification that is completely general [2]. This was done by looking for mathematical similarity, rather than features specific to music. Midi files were used for musical analysis, and very successful results were shown when grouping music into rock, jazz, or clas-

sical genres. Results were moderately successful when attempting to group pieces by composer, but got worse as sample size increased. Interestingly, the algorithm could even cluster like file types together (sorting out java class files, gene sequences from different species, and widely different styles of music). Music has also been represented through a model of rhythmic complexity [3]. With this method, rhythm was represented in a tree structure and its overall complexity was determined. Perhaps this method of representation could prove an effective way to categorize music. None of these methods have yielded perfectly successful results - even many people cannot successfully place music (especially from a short sample) into its correct genre.

# 2 Project Design

The first step of this project is to write code for reading and organizing the information stored in a midi file. A midi file is a sequence of commands preceded by how long to wait until they are executed (delta-time). For the purposes of this project, I will only need the note-on and note-off commands and their delta-time values. This will allow the code to determine which notes are playing at the same time and what rhythms are found in the piece. To sort the music, I will apply a Nave Bayes classifier. This probability model treats each probability vector as independent, but can still provide useful results. It is often used to categorize bodies of text based on what words are found within. In this case, the probability vectors are calculated based on the occurrence of words in the text. After analyzing a set of training data, the resulting model can attempt to categorize a new body of text. To apply the Bayes classifier to music, probability vectors will be calculated by analyzing the notes and rhythms a piece of music contains.

# 3 Results

So far, code can successfully input a midi file, identifying how many tracks it has and what commands they contain. The command types, arguments, and delta-time values are all stored accurately.

# 4 Discussion

This project could be a valuable tool in organizing libraries of music. Genres are often generalities that do not necessarily fit a piece exactly, but this project could determine what music really is similar and should be grouped together. It could also provide interesting insights into how we interpret music and what gives different styles their distinct feel. Another application could be composer identification. Sometimes, new manuscripts of music are discovered, but their composer is not known. Successful implementation of this project could aid in identifying the composer of any newly discovered composition.

# References


[1] Bagci, Ulas, and Engin Erzin. *Inter Genre Similarity Modelling for Automatic Music Genre Classification*. N. pag. N.p., 18 July 2009. Web. 24 Oct. 2009.

[2] Cilibrasi, Rudi, Paul Vitanyi, and Ronald De Wolf. *Algorithmic Clustering of Music*. N. pag. University of Amsterdam, 24 Mar. 2003. Web. 24 Oct. 2009.

[3] Liou, Cheng-Yuan, Tai-Hei Wu, and Chia-Ying Lee. *Modelling Complexity in Musical Rhythm*. N. pag. National Taiwan University, 26 Mar. 2007. Web. 24 Oct. 2009