

Project Proposal

Eric Liu, Zhenxuan Su

November 15, 2023

Project Title: Musical Chord Classification

Topic summary

We propose to create a model to perform binary classification of musical chords to determine whether a chord is major or minor for piano and guitar. We will train the model using a dataset of labeled audio samples of piano and guitar chords. The model will be based on a convolutional neural network (CNN) architecture, which has been shown to be effective in audio classification tasks. The CNN will take as input the spectrogram of the audio signal and output a binary classification of the chord as major or minor. (Figure 1). We will evaluate the performance of the model using standard metrics such as accuracy, F1 score. A successful model will have a high accuracy in classifying musical chords (for piano and guitar) as major or minor, which can be used in multiple applications such as musical recognition, music recommendation on music applications, and live sheet music generation.

Introduction

- The ability to accurately classify chords as major or minor can have a significant impact on the field of music and can benefit a wide range of people. It will help musicians to automatically do transcription of sheet music, which reduce the workload. Also, it will help music teachers and students to tag the mistake of playing instruments, and do auto-grade for the assignments. In the music industry, digital music providers like Spotify, Apple Music, etc., can benefit from using chord classification to recommend new songs for users based on their preferences or to recognize songs for users.
- **Note:** the most fundamental building block of music. There are 7 natural notes, represented as do-re-mi-fa-sol-la-si or A-B-C-D-E-F-G, which can be played with white keys on a piano. In between these natural notes are 5 half-step notes (e.g. A# between A and B) except B-C and E-F. They can be played with black keys on a piano. In total, there are 12 notes listed as follows and each pair of adjacent notes are half-step apart.

A A# B C C# D D# E F F# G G#

- **Frequency:** each note has a corresponding frequency (Hz) that can be calculated as

$$f = 440 \cdot 2^{d/12}$$

where d is the number of half-steps away from the reference note A.

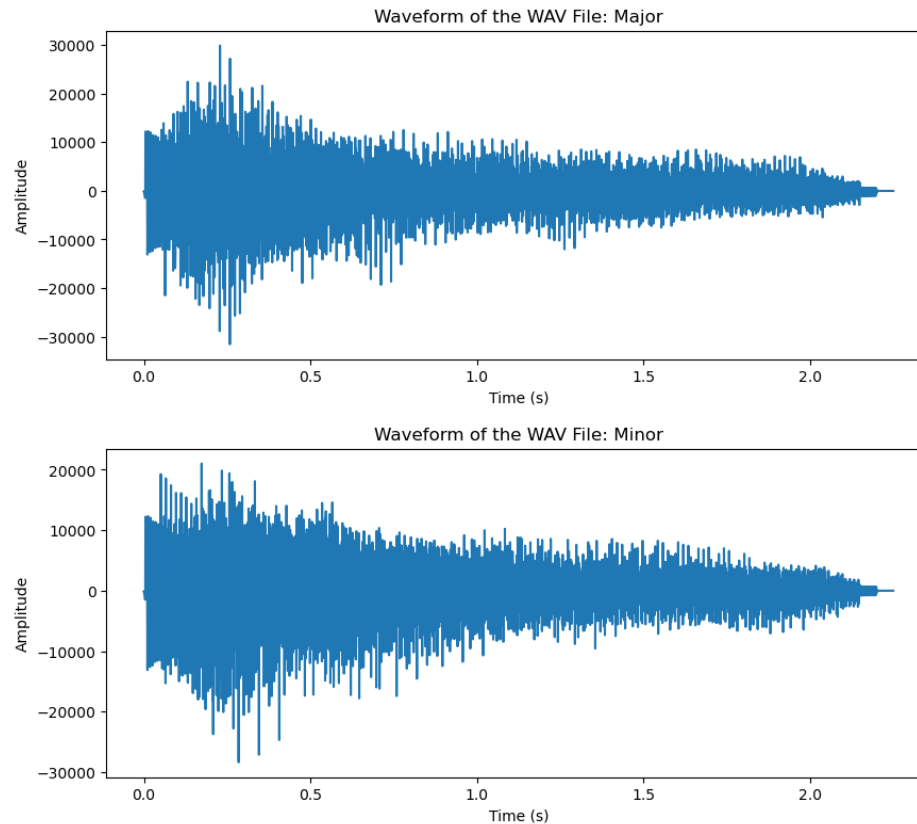


Figure 1: Waveform of musical chord.

- **Chord:** a harmonic set of notes/frequencies that are heard as if sounding simultaneously. A major chord consists of a root note, a major third (4 half-steps higher than the root), and a perfect fifth (7 half-steps higher than the root). A minor chord consists of a root note, a minor third (3 half-steps higher than the root), and a perfect fifth. For example, if the root is C, then the major chord is C – E – G and the minor chord is C – D \sharp – G.

Related Work

- Who else has worked on this problem before? There are some projects about chord classification, Ahmet Celik used a feature extraction method called voiced models to recognize chords in the music, which performed better than manual feature extraction. They also used traditional classification to create the key classifier, which was better than other classifiers.
- What did they find? The project's author, Ahmet Celik, uses the discrete and continuous Fourier transforms to extract important information from audio data. The overall accuracy is about 94
- How does the related work compare to your proposed topic? We both need to implement Fourier transforms to extract important information, however, we are trying to use CNN to solve the problem more efficiently.

Dataset description

<https://www.kaggle.com/datasets/deepcontractor/musical-instrument-chord-classification>

The dataset is 142 MB. It consists of 502 major chord files and 357 minor chord files. Each file is a single-channel wav file of 2-3 seconds with sample rate 44100Hz. We split the data into train/validation/test sets.

As the raw data is in time domain, we will preprocess it into frequency domain and experiment on models taking time domain data and frequency domain data as input, respectively and compare their performance.

Objective

The objective of this project is perform a binary classification to determine whether a chord is major or minor for piano and guitar.

We will measure model performance using two metrics, which are the accuracy and log-likelihood given by

$$L = \frac{1}{N} \sum_{i=1}^N y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i),$$

where y_i is the true label of the i-th data and \hat{y}_i is the prediction of model.

Investigation Plan

Distinguishing ideal chords is easy because they are composition of impulse in frequency domain. We will first visualize some of the data, try to perform a non-ML classification algorithm as baseline and conclude how ML can improve.

Estimated Compute Needs

Summarize the compute resources that you intend to use. We will not use a massive dataset, so PC is enough. Our team will use a Windows PC (with AMD R7, NVIDIA 2060, 16G RAM) and a Mac (with M2 chip, 8G RAM) to train the model.

Likely Outcome and Expected Results

We expect to perform the binary classification with accuracy higher than the non-ML baseline on our dataset. And we want to observe difference in performance on time-domain model and frequency-domain model and conclude reasonable interpretations.

The most possible reason for the failure is that we did not design the architecture properly and wasted all the time training it with no good outcomes.

Primary References and Codebase

We propose to build on the approach used in

- Ahmet Çelik, "[Mathematics of Music: Chord Classification](#)," Kaggle.

- Heng-Tze C., Yi-Hsuan Y., Yu-Ching Lin, I-Bin L. and H. H. Chen, Automatic chord recognition for music classification and retrieval. 2008 IEEE International Conference on Multimedia and Expo, Hannover, Germany, 2008, pp. 1505-1508, doi: 10.1109/ICME.2008.4607732.