

# 1 Introduction

## 2 Metrics for the similarity of chord progressions

### 2.1 Distance between the chord progressions

In order to evaluate how well our models emulate the chord progressions of Beethoven, we need to define a metric that quantifies the similarity between two chord progressions.

The first metric that comes to mind, is a binary metric, that just discriminates between the chord being correct or not, compared to the chord labels of the ground truth.

$$d_{bin}(x, y) = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise} \end{cases}$$

This metric, has the issue, that it treats all mistakes as equally undesirable. In the context of music, this is however not the case. For once, the model might predict the right chord, but in the wrong inversion, or it might predict a VII while we were expecting a  $V_5^6$ . It is obvious, that those mistakes are not as grave as say giving out a #IV instead of a I. The binary metric does not recognize that.

To account for this, McLeoud et. al. suggest three different metrics in [2], invoking a more contextualized approach.

*Spectral Pitch Similarity (SPS)* Is a measure to account for the perceived distance between two chords, based on the pitch content of the chords, based on psychoacoustic assumptions. For each chord label, the `eval_chord` package generates several MIDI-chords and takes a spectrogram thereof. Subsequently, for each pair of spectrograms, the cosine similarity  $\text{sim}_{\cos}(A, B) = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|}$  is calculated. The SPS distance is then given as

$$d_{SPS}(x, y) = 1 - \max(\text{sim}_{\cos}(x, y)) .$$

*Tone-by-Tone Distance* This metric treats each chord as a set of tones or pitch classes. For two chord it measures the proportion of pitch classes of chord one, which are contained in chord two and vice versa. For instance `am` and `C` share `[c]` and `[e]`, each chord consists of three pitch classes each, thus the proportion for the classes of `C` contained in `am` is  $\frac{2}{3}$ . If one chord consists of more pitch classes than the other one, the proportions are not necessary equal, i.e.

$$\text{prop}\{\text{am}, C^7\} = \frac{2}{3} \quad \text{while} \quad \text{prop}\{C^7, \text{am}\} = \frac{1}{2} .$$

In Addition to the proportions of the pitch classes, the metric also takes into account whether the root of the chords and the bass note of the chords match. Thus giving the user the option to emphasize different structural needs. The tone-by-tone

distance is then given as

$$d_{TbT}(x, y) = 1 - \frac{1}{2} \left( \frac{\text{prop}\{x, y\} + b_R \mathbb{1}_{R_x=R_y} + b_B \mathbb{1}_{B_x=B_y}}{|x| + b_R + b_B} + \frac{\text{prop}\{y, x\} + b_R \mathbb{1}_{R_x=R_y} + b_B \mathbb{1}_{B_x=B_y}}{|y| + b_R + b_B} \right),$$

where  $R_x$  and  $B_x$  are the root and bass note of chord  $x$ ,  $|x|$  is the number of pitch classes in chord  $x$  and  $b_R, b_B$  are the ‘root bonus’ and ‘bass bonus’, weights to adjust the importance of those two features.

*Mechanical distance* This metric takes into account how far the notes needed to play the chords are apart in a physical setting and on the chromatic scale. This can be useful for instance for training robotic musicians or for automatically evaluating the performance of someone on an actual instrument. Furthermore it is important to note that this metric is similar to so called voice leading distances ([3]).

As neither mechanical training nor voice leading is the focus of this paper, we will forgo the details of this metric.

As McLeoud et. al. point out, there is a correlation between the SPS and the tone-by-tone distances, especially for chords that are unrelated. The SPS metric however is more varied with the penalties for diverging. We will use both those metrics in the following discussion of the models.

## 2.2 Evaluating the quality of the labels

With the different distance metrics at hand, we can now evaluate the quality of a single estimated chord. Our goal however is to train our models on progressions of chords, thus requiring a method of applying the chord wise distances to sequences of chords.

A simple approach would be to calculate the average of the distances of the individual chords. This is also referred to Chord Symbol Recall (CSR) in [1]. While Harte mostly discusses chord recognition from audio files, the proposed metric is still applicable. The CSR is given as the average over the distances of the chords in the progression measured against the ground truth.

$$\text{CSR}(X, Y) = \frac{1}{N} \sum_{i=1}^N d(x_i, y_i),$$

where  $X = (x_1, \dots, x_N)$  and  $Y = (y_1, \dots, y_N)$  are the sequences of chords, that we are comparing.

## 3 Models

### 3.1 Simple Markov Chain Model

In order to establish a baseline, we first consider a simple Markov chain model. The model is fitted on the chords of one Beethoven string quartet. The probabilities in the matrix simply represent the frequency of the transitions between the chords. By sampling from the matrix, we can generate a new chord according to the previous one.

[print transition matrix](#)

#### 3.1.1 Performance

### 3.2 Comparaison of the Models

## References

- [1] Christopher Harte. In *Towards automatic extraction of harmony information from music signals*, 2010.
- [2] Andrew McLeod, Xavier Suermondt, Yannis Rammos, Steffen A. Herff, and Martin A. Rohrmeier. Three metrics for musical chord label evaluation. In *Forum for Information Retrieval Evaluation (FIRE)*, 2022.
- [3] Dmitri Tymoczko. Three conceptions of musical distance. In Elaine Chew, Elaine Chew, Adrian Childs, and Ching-Hua Chuan, editors, *Mathematics and Computation in Music*, Communications in Computer and Information Science, pages 258–272, 2009.