Chord Detection Using Deep Learning

{ Xinquan Zhou and Alexander Lerch } Georgia Tech Center for Music Technology

Georgia Tech Center for Music Technology
College of Architecture

Introduction

The goal of automatic chord detection is to automatically recognize the chord progression in a music recording. It is an important task in the analysis of western music and music transcription.

Several studies indicate that deep learning methods can be very successful when applied to Music Information Retrieval (MIR) tasks, especially when used for feature learning [1]. Deep learning, with its potential to untangle complicated patterns in large amounts of data, should be well suited for the task of chord detection.

Deep architectures promise to remove the necessity of custom-designed and manually selected features as neural networks should be more powerful in disentangling interacting factors and thus be able to create meaningful high-level representations of the input data.

Proposed Methods

In this work, we investigate Deep Networks (DNs) for learning high-level and more representative features in the context of chord detection, effectively replacing the widely used pitch chroma intermediate representation.

For pre-processing, we employ both time splicing and convolution (spliced filters) methods, the convolution method is shown in the following figures; for networks architectures we compare bottleneck and the common ones; for output classifiers we experiment on SVMs, HMMs and Argmax.

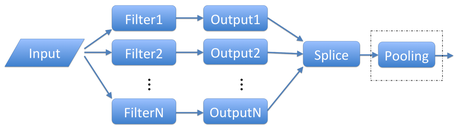

Figure 5: Spliced filters illustration

Data set

Our data set is a 317-piece collection composed of

- 180 songs from the Beatles dataset
- 100 songs from the RWC Pop dataset
- 18 songs from the Zweieck dataset and
- 19 songs from Queen dataset

The target chords are major and minor triads for every root note plus one none class resulting in 25 $(12 + 12 + 1)$ different chord labels. Ground truth time-aligned chord symbols are mapped into the major/minor dictionary:

$$Chord_{majmin} \subset \{N\} \cup \{S \times maj, min\} \quad (1)$$

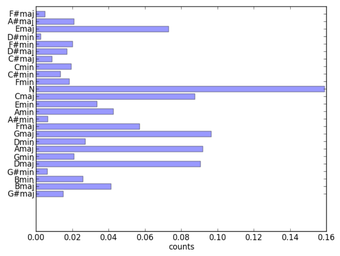

Figure 1: Chord distribution in our dataset

Results 2

Finally, we present the results of Chordino with the default settings, computed on our dataset and compare it with our system.

Method	WCSR
Chordino	0.625
Proposed (w/o max pooling)	0.919
Proposed (with max pooling)	0.916

Figure 6: Final results

Results 1

Classifier	Argmax	SVMs	HMMs
WCSR	0.648	0.645	0.755

Figure 2: Results using different post-classifiers

Three different classifiers are compared: the maximum of the softmax output (Argmax), an SVM, and an HMM. The results are unambiguous and unsurprising: the HMM with Viterbi decoding outperforms the SVM; using HMMs with a model for transition probabilities is an appropriate approach to chord detection as it models the dynamic properties of chord progressions, which cannot be done with non-dynamic classifiers such as SVMs.

The performance of both architectures is evaluated in comparison. In order to allow conclusions about overfitting, the Weighted Chord Symbol Recall (WCSR) computed with the training set is reported as well. The results show that for the three pre-processing scenarios tested: no additional pre-processing, spliced filters and spliced filters followed by a max pooling. The bottleneck architecture gives significantly better results (p = 0.023) on the test set.

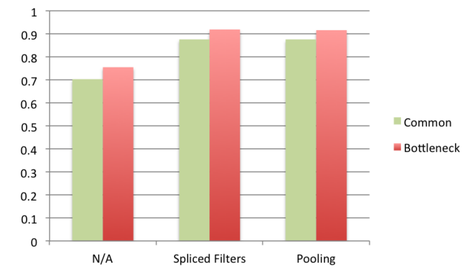

Figure 3: Results using different architectures

Conclusion

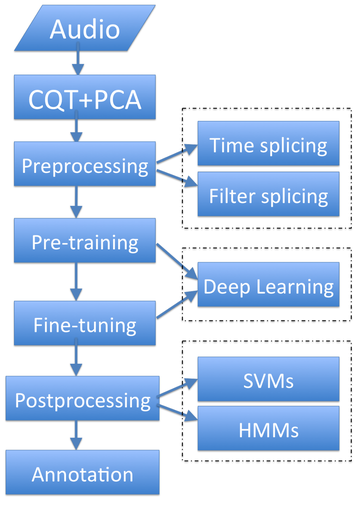

Figure 4: The flow chart of the system

- Our model is able to learn high-level probabilistic representations for chords across various configurations.
- The use of a bottleneck architecture is advantageous as it reduces overfitting and increases classifier performance.
- HMMs or Viterbi decoding algorithm fits the task better than the static classifiers.
- The choice of appropriate input filtering and splicing can significantly increase classifier performance.
- The pooling operation is preferable because it reduce the system complexity without sacrificing the performance.

References

[1] Philippe Hamel and Douglas Eck. Learning features from music audio with deep belief networks. In *ISMIR*, pages 339–344. Utrecht, The Netherlands, 2010.

Future Research

Learning a pitch class vector instead of chord likelihood by incorporating multi-label learning would allow the deep networks be independent of the number of chords to be detected. Meanwhile, instead of training chords or pitch classes we could train the output with octave independent third intervals in a multi-label scenario with 24 output nodes.

Contact Information

Email royzxq@gmail.com
Phone +1 (404) 398 7794