Introduction to Musical Corpus Studies

Release 0.0.1

Fabian C. Moss

Sep 01, 2020

CONTENT

Warning: This material is still (heavily) under construction and might change throughout the course!

You can help improving the course and let me know about any errors and inconsistencies that you find or suggest other ways of improving the course.

Welcome!

These pages present the content of the course "Introduction to Musical Corpus Studies" at the Institute of Musicology, given at University of Cologne in Fall 2020.

In the last two decades *Musical Corpus Studies* evolved from a niche discipline into a veritable research area. The growing availability of digital and digitized musical data as well as the application and development of modern methodologies from computer science, machine learning, and data science cast new light on old musicological questions and generate entirely novel approaches to empirical music research.

Moreover, the general methodological and epistemological approach of Musical Corpus Studies allows to transcend traditional intra-musicological boundaries between its sub-disciplintes (historical/systematic/ethnological/...) without sacrificing the respective specific viewpoints and perspectives.

This course offers a fundamental and practical introduction into these topics. It demonstrates, explores, and critically reflects central thematic areas and methods by means of a number of case studies. In the engagement with these topics the course also introduces elementary methods from natural language and music processing, as well as statistics, data analysis and visualization.

The course is aimed at students at the undergraduate level who have little or no empirical background and are curious about quantitative approaches to musicology.

CHAPTER

ONE

ORGANIZATION

1.1 Overview

No.	Date	Time	Room[1]	Topic	Methods
1	Fr., 13.11.2020	16:00-17:20 Uhr	B	Introduction / Background	
2		17:40-19:00 Uhr		Folk Songs, Melodies, Pitches and Intervals	frequencies, mean, variance
3	Sa., 14.11.2020	09:00-10:20 Uhr	B	Jazz Solos, Melodies	Regular Expressions
4		10:40-12:00 Uhr		Beethoven's string quartets Harmony	n-grams, Markov models
		12:00-13:00 Uhr		Lunch Break	
5		13:00-14:20 Uhr		Pop Charts Billboard 100, harmony,	Clustering, k-means, [Hidden Markov Models]
6		14:40-16:00 Uhr		Free group work	
7	Fr., 11.12.2020	10:00-11:20 Uhr	A	Brazilian Choro, harmony, form,	Context-Free Grammars
8		11:40-13:00 Uhr		19th century piano music, harmony	Probabilistic CFGs
9	Sa., 12.12.2020	09:00-10:20 Uhr	B	Malian Percussion Music, rhythm, meter	
10		10:40-12:00 Uhr		Electronic Music 1950-1990	
		12:00-13:00 Uhr		Lunch Break	
11		13:00-14:20 Uhr		Free group work	
12		14:40-16:00 Uhr		Recapitulation and conclusion	

[1] A: Alter Seminarraum 1.408; B: Neuer Seminarraum 1.315

1.2 Credits

Important: Ich gehe in der Seminarplanung von 12 Semesterwochen à 2 SWS aus, für das gesamte Blockseminar also 24 SWS. Das Seminar wird mit 3 CP bewertet, was 90 Stunden aktiver Arbeit entspricht. Davon entfallen 24 SWS an die Präsenzzeit im Seminar plus 48 SWS an Vor- und Nachbereitung der Seminarsitzungen. Die verbleibenden 18 SWS sind für die Lektüre der Fachliteratur vorgesehen.

CHAPTER

TWO

INTRODUCTION

2.1 About this course

2.1.1 About me

- Music and Mathematics education (Uni & HfMT Köln)
- MA Musicolgy (HfMT Köln)
- PhD Digital Humanities (EPFL)

2.1.2 Focus of this course

Programing introductions often boring. A lot of time lost in introducing basic concepts and techniques (important!) but quite remote from actual (!) applications. Examples are usually "toy examples" that work well, but the transition to real-world applications is difficult. Of course, the example studies discussed in this course work well, too. However, they are without exception taken from peer-reviewed, published, open access articles. They thus reflect actual, recent research questions that reflect current research.

This course takes thus the opposite approach to "toy examples". We will not introduce many specific programing concepts. The course rather showcases what is possible with musical corpus studies. If this sparks your interest, it will be much easier to pick up the basics for yourself, knowing what they are *for* and being motivated intrinsically. If you are not particularly interested in doing this kind of work yourself, you will still see a broad range of applications that are much more useful to you than learning (or not learning) Python basics.

2.2 What are Musical Corpus Studies?

tbc... (text from diss?)

2.3 Epistemological goals

tbc...

2.4 Issues

tbc [**?**][**?**][**?**][**?**][**?**][**?**][**?**][**?**][**?**]

2.5 MCS and traditional musicology

tbc

CHAPTER

THREE

FOLK SONGS AND THE MELODIC ARC

Huron. . . / MusThe Tutorial

Table 3.1: CSV-TABLE TITLE

col1	col2	col3
val11	val12	val13
val21	val22	val23
val31	val32	val33

CHAPTER

FOUR

SOLOS IN THE *WEIMAR JAZZ DATABASE*

Fig. 4.1: Photo by Janine Robinson on Unsplash

The first project we will have a look at is the Jazzomat project. Transcriptions of Jazz solos [**?**]. The *Weimar Jazz Database* (WJD) consists of 456 transcriptions of Jazz solos from diverse substyles. As all the corpora that we deal with here, it is freely available on the internet.[1]

The WJD contains a number of tables:

[1] https://jazzomat.hfm-weimar.de/dbformat/dboverview.html

Table 4.1: Tables in the *Weimar Jazz Database*

Table name	Description
`beats`	Table for beat annotation of WJD melodies, referenced by `melody(melid)`
`composition_info`	Infos regarding the underlying composition of a WJD solo, referenced by `melody(melid)`
`db_info`	Information regarding the distributed database file like version information, license, etc
`esac_info`	EsAC infos for EsAC melodies, referenced by `melody(melid)`
`melody`	Main table for all melody events
`melody_type`	Indicated type of melody: WJD solos or EsAC (Folk songs using Essen Associative Code), referenced by `melody(melid)`
`popsong_info`	Pop song infos, referenced by `melody(melid)`
`record_info`	Infos regarding the specific audio recording of a WJD solo was taken from, referenced by `melody(melid)`
`sections`	All sections (phrase, chorus, form, chords, etc.), referenced by `melody(melid)`
`solo_info`	Solo infos for WJD solos, referenced by `melody(melid)`
`track_info`	Information specific to a track on a record (or CD)
`transcription_info`	Transcription infos for WJD solos, referenced by `melody(melid)`

Here, we focus on the main table `melody`. First, we download the entire database from https://jazzomat.hfm-weimar.de/download/download.html (under "Weimar Jazz Database") and save it as the file `wjazz.db`.

```
import sqlite3 # for working with databases
import pandas as pd # for working with tabular data

# create connection to database
conn = sqlite3.connect("wjazzd.db")

# read all entries of the `melody` table into a pandas DataFrame
df = pd.read_sql("SELECT * FROM melody", con=conn)

df.head()
```

```
>>> df.head()
output
```

The part of the code `SELECT * FROM melody` reads "Select all entries from the table 'melody'".

CHAPTER

FIVE

HARMONY IN BEETHOVEN'S STRING QUARTETS

5.1 Access the data

The data lies on the GitHub repository DCMLab/ABC. Either download the `.tsv` file directly and open it in pandas or load it from the URL as follows:

```
import pandas as pd

df = pd.read_csv("https://github.com/DCMLab/ABC/corpus.tsv", sep="\t")
```

The corpus is now stored in the variable `df`.

5.2 Harmonic Annotations

- regular expressions

[?][?]

5.3 Chord Transitions

- n-grams

CHAPTER

SIX

HARMONIC CLUSTERS IN POP CHARTS

Clustering analysis in [**?**].

```
def zipf_mandelbrot(x, a, b, c):
    """Zipf-Mandelbrot function of `x` given parameters `a`, `b`, `c`."""
    z = a / ( (b + x)**c )
    return z
```