Lab 3 – MATH 240 – Computational Statistics

Caroline Devine
Colgate University
Math Department
cdevine@colgate.edu

2/11/2025

Abstract

This lab is an extension of Lab 2 which focused on how to automate data extraction for .WAV files through batch processing as well as extracting and analyzing musical features from `.JSON` data. The extension is outlined in Task 3 under the methods section which compiles data from multiple tracks using the Essentia model and LIWC text analysis. The goal is to use the methods below to create data frames that can be used to facilitate analysis of musical influence.

Keywords: Data Collection, Lists, Batch Files, For Loops, JSON package

1 Introduction

The three bands: The Front Bottoms, Manchester Orchestra, and All Get Out collaborated on a song called "Allentown"(Ross, 2018) in 2018. This project aims at answering the research question of which band contributed or imprinted most to this song. To analytically determine this, we will analyze 180 tracks and "Allentown" to examine the data about what each band sounds like. These music files have metadata, key musical features, that can be analyzed to identify distinctions between the influence of these bands. The goal is to learn how to install, load, and learn how to use libraries; work with character objects; code `for()` loops; and access elements of vectors and lists.

2 Methods

To analyze data sets of track files from different artists and albums, we built a batch file for data processing, extracted key music features from a .JSON file, and complied data from a larger set merging them with the Essentia models and LIWC text tool analysis.

2.1 Task 1: Building a Batch File for Data Processing

The dataset used was a directory called "Music" containing two artists: OfficeStuff and PeopleStuff. The first step in analyzing the provided .WAV files was to build a batch file was installing the `stringr` package which allows for string manipulation when extracting information out of the directories, subdirectories, and track files in the Music folder.

Steps to batch file:

1. Retrieve the albums from Music folder using `list.dirs()` function
2. Using `stringr` functions to isolate and count .WAV files for each of the albums
3. Extract track number, artist name, track title, album name from track file names and directory
4. Create command lines for each .WAV file to convert into the naming convention for .JSON files
5. Save the converted .JSON files to a text file called `batfile.txt`

The `for()` loop function was used to automate the system for more efficient processing and future use.

2.2 Task 3: Compiling Data from Essentia

Using a `for()` loop, we extracted eight key musical features from a singular song "Au Revoir (Adios)" in the album Talon of the Hawk by The Front Bottoms using the uploaded .JSON file. The eight features are the overall loudness, dissonance, pitch salience, tempo in bpm, beat loudness, danceability, and tuning frequency.

By iterating over a list of 181 .JSON files from the Essentia model (Bogdanov et al., 2013), we extracted the same eight key musical features from each track. We stored the data in a data frame with the corresponding artist, album, and track names.

2.2.1 Load and Clean Data from Essentia Models

Using the provided .csv file, `EssentiaModelOutput.csv`, we used three datasets: DEAM, emoMusic, and MuSe to average each of their valance and arousal. New columns were created using `rowMeans()` of key musical features using different extractors such as Discogs-EffNet and MSD-MusiCNN in a similar manner as the valence and arousal columns. We renamed a column for clarity purposes changing `eff_timbre_bright` to `timbreBright`. Lastly, we isolated the desired columns which included the features created, renamed, and the artist, album, and track columns. This removed the unnecessary columns for future efficient analysis, leaving a clean data frame.

2.2.2 LIWC Text Analysis Tool

To analyze the lyrics of the tracks, we utilized a text analysis tool called LIWC which provides features that describes thoughts, feelings, and personality traits based on the language used. We loaded the `LIWCOutput.csv`. We merged the data from the **streaming_music_extractor** calls, the Essentia models, and the LIWC into one data frame using `merge()` function. The three data frames merged are called `df, cleaned.essentia.model, and LIWCOutputdf`. The resulting data frame consisted of 181 rows and 140 columns ensuring no column duplication or omission. Additionally, we renamed `function.` to `funct` because using `function` as a column name can result in issues in coding within `R`.

Lastly, we wrote two .csv files with one containing all tracks except "Allentown" called `trainingdata.csv` and the other containing only "Allentown" called `testingdata.csv`. This is useful to evaluate the information solely based on "Allentown" as the initial research question calls for.

3 Results

For Task 1, we successfully analyzed .WAV files, storing the results in .JSON files. For Task 2, we collected the correct loudness, energy, danceability, tempo, key, mode and duration for the provided track's json file output. Task 3 extracted key musical features from 181 tracks,

cleaned and organized the data from Essentia model's .JSON outputs, combined those data points with LIWC text analysis, and created training and testing data sets(Bogdanov et al., 2013).

5 Table

4 Discussion

This process is now automated and can be replicated with other files as well. This creates an efficient way to compare larger data sets of multiple artists or albums. The integrated, structured data frame allows for specific analysis to address research question in future work.

References

Bogdanov, D., Wack, N., Gómez Gutiérrez, E., Gulati, S., Boyer, H., Mayor, O., Roma Trepat, G., Salamon, J., Zapata González, J. R., Serra, X., et al. (2013). Essentia: An audio analysis library for music information retrieval. pages 493–498.

Ross, A. R. (2018). Manchester orchestra and the front bottoms are finally together on "allentown". *Vice*.

artist	feature	description
All Get Out	spectral_rolloff	Out of Range
Manchester Orchestra	spectral_rolloff	Within Range
The Front Bottoms	spectral_rolloff	Out of Range
All Get Out	dissonance	Outlying
Manchester Orchestra	dissonance	Within Range
The Front Bottoms	dissonance	Out of Range
All Get Out	average_loudness	Outlying
Manchester Orchestra	average_loudness	Within Range
The Front Bottoms	average_loudness	Outlying
All Get Out	chords_strength	Outlying
Manchester Orchestra	chords_strength	Within Range
The Front Bottoms	chords_strength	Out of Range
All Get Out	conj	Out of Range
Manchester Orchestra	conj	Outlying
The Front Bottoms	conj	Within Range
All Get Out	Perception	Out of Range
Manchester Orchestra	Perception	Within Range
The Front Bottoms	Perception	Out of Range
All Get Out	OtherP	Outlying
Manchester Orchestra	OtherP	Outlying
The Front Bottoms	OtherP	Within Range
All Get Out	positivewords	Outlying
Manchester Orchestra	positivewords	Outlying
The Front Bottoms	positivewords	Within Range

6 Appendix

Below is the table of the 20 features that we identified as features that differentiate the artists influence on the song.

artist	feature	description
All Get Out	spectral_skewness	Outlying
Manchester Orchestra	spectral_skewness	Within Range
The Front Bottoms	spectral_skewness	Out of Range
All Get Out	spectral_rolloff	Out of Range
Manchester Orchestra	spectral_rolloff	Within Range
The Front Bottoms	spectral_rolloff	Out of Range
All Get Out	spectral_kurtosis	Outlying
Manchester Orchestra	spectral_kurtosis	Within Range
The Front Bottoms	spectral_kurtosis	Out of Range
All Get Out	spectral_entropy	Outlying
Manchester Orchestra	spectral_entropy	Within Range
The Front Bottoms	spectral_entropy	Out of Range
All Get Out	spectral_energyband_middle_high	Out of Range
Manchester Orchestra	spectral_energyband_middle_high	Within Range
The Front Bottoms	spectral_energyband_middle_high	Out of Range
All Get Out	spectral_complexity	Out of Range
Manchester Orchestra	spectral_complexity	Within Range
The Front Bottoms	spectral_complexity	Out of Range
All Get Out	spectral_centroid	Out of Range
Manchester Orchestra	spectral_centroid	Within Range
The Front Bottoms	spectral_centroid	Out of Range
All Get Out	melbands_spread	Out of Range
Manchester Orchestra	melbands_spread	Within Range
The Front Bottoms	melbands_spread	Out of Range
All Get Out	melbands_flatness_db	Out of Range
Manchester Orchestra	melbands_flatness_db	Within Range
The Front Bottoms	melbands_flatness_db	Out of Range
All Get Out	erbbands_skewness	Out of Range
Manchester Orchestra	erbbands_skewness	Within Range
The Front Bottoms	erbbands_skewness	Out of Range
All Get Out	erbbands_flatness_db	Outlying
Manchester Orchestra	erbbands_flatness_db	Within Range
The Front Bottoms	erbbands_flatness_db	Out of Range
All Get Out	dissonance	Outlying
Manchester Orchestra	dissonance	Within Range
The Front Bottoms	dissonance	Out of Range
All Get Out	barkbands_skewness	Out of Range
Manchester Orchestra	barkbands_skewness	Within Range
The Front Bottoms	barkbands_skewness	Out of Range
All Get Out	barkbands_flatness_db	Outlying
Manchester Orchestra	barkbands_flatness_db	Within Range
The Front Bottoms	barkbands_flatness_db	Out of Range
All Get Out	average_loudness	Outlying
Manchester Orchestra	average_loudness	Within Range
The Front Bottoms	average_loudness	Outlying
All Get Out	chords_strength	Outlying
Manchester Orchestra	chords_strength	Within Range
The Front Bottoms	chords_strength	Out of Range
All Get Out	conj	Out of Range
Manchester Orchestra	conj	Outlying
The Front Bottoms	conj	Within Range
All Get Out	Perception	Out of Range
Manchester Orchestra	Perception	Within Range
The Front Bottoms	Perception	Out of Range
All Get Out	OtherP	Outlying
Manchester Orchestra	OtherP	Outlying
The Front Bottoms	OtherP	Within Range
All Get Out	positivewords	Outlying
Manchester Orchestra	positivewords	Outlying
The Front Bottoms	positivewords	Within Range