BACLaudio: Music Sentiment Analysis Visualization

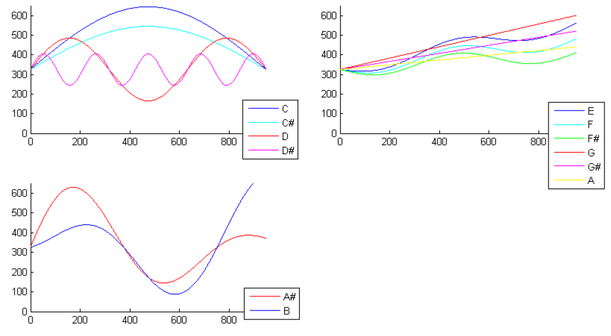
Becca Patterson, Audrey Lewis, Cecilia Diehl, Liz Sundsmo
Software Design Spring 2016, Olin College of Engineering

Features

To analyze the piece of music we look at the chords, bpm, and lyrics. To represent these three values, a dot is released on the beat. The color of this dot represents the lyric sentiment, red/orange if positive and blue if negative with a gradient between the two. If the lyrics are completely neutral the dot is colored grey. The dot follows a path across the screen representing the key of the song that the dot was released on. Since there are twelve possible chords there are twelve unique functions. Each dot and function is produced with a mirrored duplicate to improve visual appearance.

The song itself is determined by the user. Our wrapper allows the user to determine what song they wish to analyze as long as they have access to the correct files. After the song is analyzed and played once it is pickled into a folder with the other necessary files and saved. This means that if the same song is played more than once the analysis does not need to occur again.

Function Graphs



Color Key

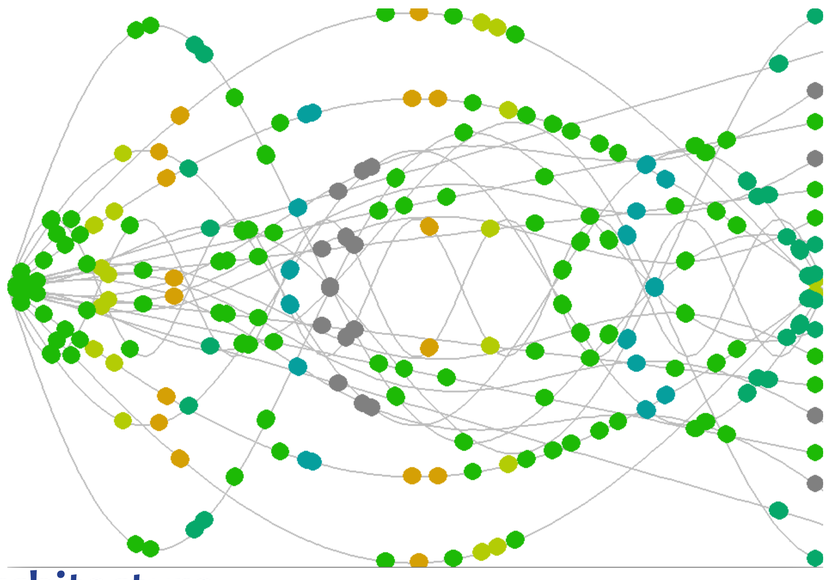
The color key maps to the lyric sentiment. A line that is completely positive will be red, and completely negative will be dark blue. Any other ratio of positive to negative exists on our gradient scale. Most of the colors that will be seen in a visualization of a song are a greenish-blue to orangish-red because it is rare that a line is only positive or only negative. If there are no lyrics or the sentiment is not polar enough, the color is set to a default gray.

What is BACLaudio?

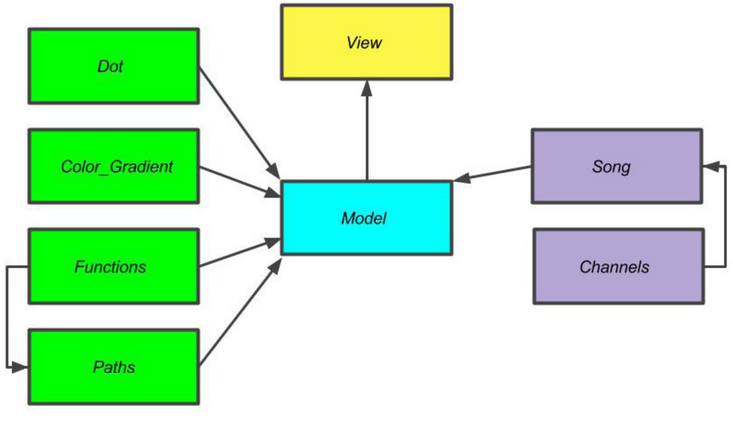
BACLaudio is an auditory and visual experience generator. Taking in a variety of songs (we've chosen to theme this iteration on the Shrek soundtrack), BACLaudio generates a visualization of the lyrical and instrumental sentiment. In the form of a series of dots and functions, the continuously running patterns depict the instrumental sentiment based on chords and beat. Meanwhile, the color of the dots is determined by the sentiment of the song lyrics which is determined by language processing.

We decided to pursue this project because we had all fallen victim to rocking out to happy tunes, only to be struck by the later realization that the lyrics were really negative, changing our entire perception of the song. We were curious about creating a visualizer that would allow us to see these discrepancies in mood.

Display



Code Architecture



Pitfalls

Currently the user must wait several seconds for the program to complete analysis of the song and play, which creates a less than ideal audio and visual experience.

We were initially going to identify key and correlate with a sentiment; for example, major to happy and minor to sad. However the library we were using didn't have clear enough documentation and the API we planned to use as back-up was purchased by another company and put offline. Instead we have identified what chord is being played, but we have not found a way to map a single chord to a sentiment. So the chord determines the function path, but is not indicative of a mood. Currently we do not make the relationship between the function path and chord clear in our visualization, which could improve the experience of our users.

Future Work

The next logical step in code implementation would be to 'chunk' the audio and text so that we can simultaneously perform analysis and display the visualization. In addition we would attempt to make the connections between visuals and information more clear. After that, we would aim to increase our code flexibility and increase our song library so users have more song options.

Attributions and References

Time - Reading, checking timestamps, timing
Numpy - Math
Librosa - Obtaining beats, music chord changes
Datetime - Timestamping events for channel classes
Pygame.mixer - Use of music with Pygame
Urllib2, json - Interaction with Natural Language Toolkit (NLTK) text sentiment API

More information about how to obtain and use BACLaudio can befound on our GitHub repository: https://github.com/audreywl/baclaudio