Fall 2021 CS 687 Capstone Project
Final Report
Meaningful Classification for Music Recommendation Systems

Brian Geving
Advisor: Sion Yoon
MS in Computer Science
School of Technology and Computing (STC)
City University of Seattle (CityU)
gevingbrian@cityu.edu, yoonhee@cityu.edu

Abstract

While the field of Music Information Retrieval (MIR) is steadily progressing, the majority of music recommendation systems still operate using collaborative filtering. Collaborative filtering operates by finding patterns between music represented in collected playlists. While this method works well for music that has a large amount of representation, it is a very poor method for recommending music that has very little availability. By utilizing machine learning classification algorithms and low-level audio feature extraction, there exists a method to classify music with specific parameters. Instead of using modern music genres as a way to relate music, using labels provided by collaborative filtering can provide a high accuracy way to recommend similar music with low representation.

Keywords: Machine Learning, SVM, Collaborative Filtering, Classification, Music, Librosa, MIR

1. INTRODUCTION

The sheer volume of music being put out in the past decades has increased exponentially due to the ease and lower costs to create and self-distribute music. In 2020, it was said that over 40,000 new tracks were uploaded every day (Miller, 2020). In order to define how this music sounds and relate it to other music, we typically use music genres. Music genres are largely classified by humans, with even more of it being done by the artists themselves. This leads to many more genres and subgenres being created to specify how a particular song sounds. Many times a song will be defined by multiple genres and subgenres.

With the help of music visualization libraries, being able to classify music with machine learning algorithms results in a song with one label that can more objectively classify the sound of a song. With sufficient testing, machine learning classification can create music recommendation software with stronger musical relationships.

Problem Statement

While the field of Music Information Retrieval (MIR) is expanding, many music recommendation software and applications are still based solely on the collaborative filtering of tags and playlists (Vall et al. 2019) rather than any use of audio feature extraction. How can MIR and classification be used in a way to create a strong music recommendation system?

Motivation

Modern music genres do not exist solely as a means to group music by sound. In fact, as many as one-third of today's music genres are determined by a scene or industry (Lena & Peterson, 2008).

Many musicians do not want to be bound to a genre, however, having these genres is necessary for both consumers and those distributing the music. One example is that of Heavy Metal, where not only fans but the musicians themselves often debate what music fits within the genre and what the genre itself entails (Lena & Peterson, 2008).

These debates, while interesting, are counter-intuitive to "grouping" music and giving music recommendations. Previous studies have shown that music can be analyzed by a computer and even given an emotion descriptor based on musical features (Han et al., 2010). Being able to have a more objective way of analyzing music will hopefully lead to creating a recommendation system with strong similarities.

Approach

Rather than use genre as a classifier label, songs should be separated into groups based on the similarity of sound. To determine sound similarity, online music databases that typically function with collaborative filtering, such as last.fm will be used to determine similar/recommended songs.

After all selected songs have been given a label, a classifier algorithm and audio features will be selected based on what works best for the model. All audio features will be extracted using the Python library librosa and most of these features will be based on Mel-frequency cepstral coefficients (MFCC).

Conclusions

Usings labels provided by groups with collaborative filtering, there exists more than enough song data to train a classification model. By combining these two approaches, the end result should be a high accuracy system that covers the weak points in a typical collaborative filtering recommendation system.

2. BACKGROUND

The majority of music available by volume has a small listening base. Music that has very little online presence is poorly represented by recommendation systems that use collaborative filtering (Vall et al. 2019). The issue here is that while collaborative filtering may result in a strong music recommendation system for popular artists, the system crumbles for the majority of available music.

Using genres is the most common way to group like-sounding music. The problem with using music genres is that the information it provides can be too broad, misleading, or irrelevant to its sound. Fans and the artists themselves may have differing opinions on what genre a piece of music might be, and even in cases where the genre is agreed upon, a large portion of music genres describe a scene or industry rather than the musical sound (Lena & Peterson, 2008).

The current challenge then is to find a recommendation system that can work based on the sound of the music itself. This makes both the availability of music irrelevant to the recommendation and just requires a one-time classification. The issue is that there is no universally agreed-upon metric to measure music by sound.

3. RELATED WORK

The literature review is provided as a means to discuss currently available classification methods as well as finding a balance of categories that adequately describe a wide range of music.

Literature Review

There are many methods of classification, and the choice of which to use can greatly impact the result of the project. In order to pick a method of classification that will result in the strongest agreement, we must first know how our classification will be applied.

The first question that must be asked is whether our classification is binary or multi-class. With the decision to separate our categories into multiple groups, we are clearly using multi-class classification. This knowledge alone makes certain classification methods unusable or unappealing. For example, simplistic support vector machine (SVM) approaches are defined by dividing data points into 1 or 0, making its native use only applicable for binary classification (Aggarwal, 2015). Other approaches, while viable, may result in poor accuracy. k-Nearest Neighbors (k-NN) can be easily applied to a multi-class classification problem, however the result gives at most twice the error rate as Bayes probability (Aggarwal, 2015).

For multi-class classification, our methods available can be broadly separated into two groups - algorithms that can natively apply multi-class applications and methods that involve decomposition to a collection of binary classifiers (Li et al., 2003). k-NN discussed earlier would fit in the former approach, while SVM would fit in the latter. Sadly, this doesn't exactly reduce our classification methods overly much.

Luckily, there have been multiple studies on audio classification. Many of these studies conducted for low-level audio classification used SVM as a conventional classifier (Pandeya & Bhattarai, 2021, Li et al. 2003, Vall et al. 2019, Han et al. 2010, Murthy & Koolagudi, 2018, Wang, 2020). One study (Li et al., 2003) found that the overall accuracy of using an SVM classifier was over 5% more accurate than

Linear Discriminant Analysis (LDA) for classifying music into broad genres and more than 10% more accurate than k-NN. Another study (Wang, 2020) found that an SVM classifier had both higher accuracy and more efficient computation time than a backpropagation neural network (BPNN) approach.

Classification algorithms aside, knowing how to categorize music is underappreciated. While most people will refer to music by their genre, musical genres are not an exact science which is why classification of them is difficult in the first place.

The largest difficulty when using musical genres to classify music is that a musical genre may have little to nothing to do with the actual sound. In fact, as many as one-third of used musical genres have more to do with the scene or industry it came out of (Lena & Peterson, 2008). Some examples of this include Classic Rock and New Wave of British Heavy Metal (NWOBHM). The former refers to a period of time, and the latter refers to a time and place. It can be argued that these genres have their own unique sound as well, however they must meet other requirements in order to fit within the boundaries of the genre.

One of the ways to get past this hurdle is to not use musical genres in the first place. Some past studies (Pandeya & Bhattarai, 2021, Han et al., 2010) have decided to classify music based on its emotive sound. While deciding what emotions certain sounds evoke is another difficult topic to get into, it at least resolves the problem of having to extract possibly unavailable features such as time and place recorded.

Another way to classify music and sounds is to use only broad genres. There have been high accuracy studies (Li et al. 2003) classifying music by these broad genres. While being a terrific finding, classifying genres in broad terms has varying usefulness depending on the application.

Music recommendation applications are popular and have varying degrees of success. A common way to recommend songs in a playlist is to use collaborative filtering in order to find other playlists that have the current song. This approach, however, is not very useful on songs that only appear on very few playlists (Vall et al., 2019). Similarly, classifying using broad genres is not very helpful to someone looking for a more specific sound.

The use of Mel-frequency Cepstral Coefficients (MFCC) is a way to take a representation of an audio clip based off of human perception (Singh & Rajhan, 2011). Because of this basis on the mel-scale and human based perception of acoustic sound and frequency, it is very commonly used as a way to recognize, verify, or reconstruct human speech (Milner & Shao, 2006).

When it comes to music, like speech, human perception is all that matters. It's for this reason that MFCC also has many applications in the field of Music Information Retrieval (MIR) (Loughran et al., 2008).

Review Conclusions

Studies on low to mid-level audio feature extraction show that SVM classifiers tend to have the best performance. Studies also show the difficulty of using classifiers on music while having meaningful and applicable results.

The field of music classification has been well studied. Due to limitations of past classification approaches, music recommendation systems based on classification have been largely unsatisfactory.

4. APPROACH

To use machine learning in order to cover a broad spectrum of music requires a multi-class classification approach. For multi-class classification to perform well, it needs both an algorithm well suited to the classification as well as enough appropriate features to give adequate results. In our case, the algorithm used for classification will be SVM, and the features will be extracted from the music itself.

User Requirements

To develop a classification method that covers a broad enough spectrum of music to give recommendations based on any song requires:

1. Enough labels to classify all different types of music.
2. A way to create or generate labels that group similar sounding music.
3. A classification algorithm that fits the model.
4. An adequate amount of correctly labeled songs to train the classification model.
5. Features that correspond to desired classification results.

Design

The recommendation system proposed will function as a library of music with individual songIDs being grouped together by their sound

classification. The library will only contain music that has been given a classification. In order to obtain a classification, a song must enter a one-time analysis where features will be extracted from the audio file and be given a label using those features.

In order for songIDs to be unique, there must be some source to differentiate songs on a basis other than simple song metadata such as album, artist, song name, etc. Songs hosted by Spotify would meet the requirements by using song URLs as unique songIDs.

Testing done within this project is not large enough in scope to bother with many of these requirements. For testing purposes, songs grouped by their classified sound will use their song name as identification.

Implementation

A .csv file of all pertinent data will be created by looping through all songs in a folder that will serve as the song database. Songs in the database will be separated into subfolders that will represent their sound classification.

Each song will serve as one entry in the .csv file and contain the song's label, the song name, and all features used for classification purposes. When all data is successfully extracted, a dataframe can be created to train and test the classification method.

Technologies Used

Scikit-learn is a python-based machine learning library featuring various classification algorithms, including the one-vs-one SVM classifier that will be predominantly used.

Librosa is a python-based library used for audio analysis. The library includes ways to extract features of pitch, tempo, and others directly from audio files. All features used for classification will be obtained directly from the feature extraction tools available in the librosa package.

Pandas and NumPy are widely used data science libraries for Python. NumPy is a library that focuses on multi-dimensional arrays, while pandas, built on NumPy, is used to create dataframe objects.

SciPy is a scientific python library that will be only used for its skew() function in this project.

Matplotlib is a python library used for data visualization. Figures of data in this paper will utilize Matplotlib.

5. DATA COLLECTION

Two different music databases will be used. The first will be the very commonly used GTZAN dataset. The GTZAN dataset contains 1,000 30 second clips of songs that are evenly distributed into 10 genres; blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, and rock. This dataset is used very often in order to create classification systems according to genre. In this case, because of its availability and size, it will be used as a comparison group and for measuring model accuracy.

The second selected music database will be hand-picked in order to be closer to the recommendation system desired. The dataset contains 60 songs evenly divided into 5 different groups. Each group will be based off of one song, and the similar/recommended songs for it from the website last.fm. The sound of the song groups is far apart enough where one song does not share a similarity with a song in another group. The idea is to see if a classification system based solely off of audio features can recreate the same recommendations as a collaborative filtering system.

All songs in the second dataset had 30 second clips extracted at an offset of 30 seconds into the song. Before this extraction, the songs had their entire song duration recorded as a floating point number of seconds. The reason only 30 seconds were used for the feature extraction of each song is to normalize the amount of samples taken from every song. Samples taken from a song for all used audio features is given by the formula

□□□□□□ = □□□□□□□□ * (□□□□□□ □□□□ / □□□ □□□□□□)

Librosa's default sample rate and hop length are 22050 and 512, respectively, giving 1292 frames taken for every audio feature for both datasets.

All songs used in both datasets had 5 features extracted: Mel-frequency Cepstral Coefficients (MFCC), Spectral Centroid, Zero Crossing Rate (ZCR), Chroma Frequencies, and Spectral Roll-off. In addition, the second hand-picked dataset has an additional feature that contains the unaltered song duration. This feature should have an impact on song similarity. However, this feature is impossible to use for the GTZAN dataset due to how it has already shortened all songs to 30 seconds before feature extraction. All features extracted result in an array of 1292 floating point numbers with MFCC and song duration being the exceptions. The MFCC was a 2d array based on

the first 15 coefficients represented by the shape (15, 1292), and the song duration is represented by a single floating point value.

In order to transform all feature arrays into a single floating point number, multiple statistics were performed on each. First, the 2D MFCC array was transformed into 15 1D arrays. Each feature array of 1292 elements then had the following elements derived from them; mean, standard deviation, skew, maximum number, median, and minimum number.

The output .csv files contained 114 features for the GTZAN dataset and 115 features for the handpicked dataset, excluding the label and song name.

6. DATA ANALYSIS

All features extracted, with the exception of song duration, contained an array of 1292 frames with floating point number results. For the purposes of data visualization, figures will contain the original output before statistics were performed and dimensionality reduced.

Mel-frequency Cepstral Coefficients (MFCC)

MFCCs are a way of giving spectral representation to an audio signal. The amount of coefficients that actually give impactful values for music is up for debate. One study (Loughran et al. 2008) noted that 12 coefficients were generally used for speech, and 15 coefficients were the best value for instrumentation. We will be using the first 15 coefficients as representation in light of these findings.

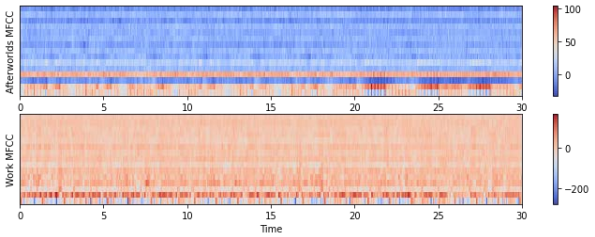

Fig 1: MFCC of two songs

The MFCC coefficients contain rate changes in spectrum bands. If a coefficient has a positive value, the spectral energy is concentrated in low frequencies, and if the value is negative, the reverse is true. Figure 1 contains an MFCC comparison of two different songs from two different genres. The song in the above plot is from a metal song that contains no vocals in the 30 seconds used, while the below plot contains a rap song.

Spectral Centroid

The spectral centroid is the weighted mean of the frequencies present in an audio signal. Each frame contains the result of averaging the 512 samples used as librosa's default hop length.

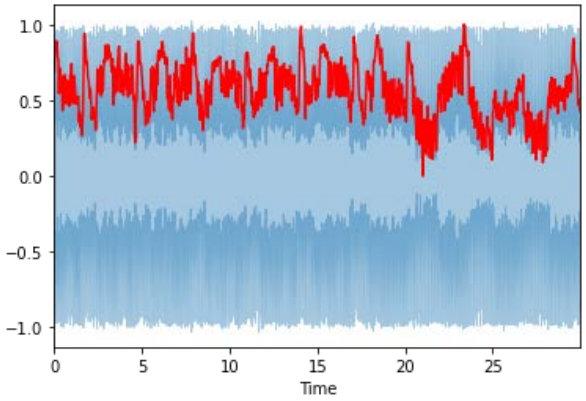

Fig 2: Spectral Centroid of Song1

Figure 2 shows a normalized plot of the spectral centroid for the above song shown in Figure 1. The red lines are representative of the spectral centroid, while the blue is the waveform for the song itself. Using solely the mean of all spectral centroid frames has little merit, however generalized trends tend to be different for different genres.

Zero Crossing Rate (ZCR)

The ZCR is the rate that an audio signal changes signs, from positive to negative or vice versa, in each frame. ZCR is a good measurement of the "noisiness" of an audio signal, and in the case of music, it can also be a good indicator of the amount of percussion used in a song.

Chroma Frequencies

A chromagram is a simple representation of the pitch of the song at a certain point in time. The chroma feature breaks the spectrum into 12 bins representing the 12 semitones in an octave.

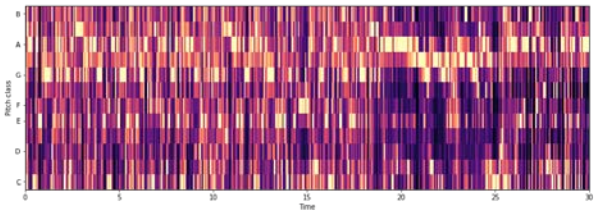

Fig 3: Chromagram of Song1

Figure 3 displays the chromagram for the song used in the previous figures. The y-axis represents the pitch class/semitone/chroma of the song while the x-axis represents the time.

Spectral Roll-off

The roll-off frequency is defined as the frequency under which some specified percentage of spectral energy is contained. The default roll_percent for librosa is 85%. The roll-off is used to distinguish between harmonic and noisy sounds. Harmonic sounds are below the roll-off, while noisy sounds are above roll-off.

Song Duration

The song's duration is the simplest feature to analyze. By itself, the feature holds little weight and could be used for any recording. However, certain musical genres tend to have trends based on song length. For example, punk songs tend to have shorter songs around the two minute mark, while songs by post-rock artists can often reach over ten minutes long.

Especially when it comes to song recommendation, the song's length can be an important indicator of what a user is looking for.

7. FINDING

When evaluating model performance on various SVM kernels, the rbf model gave the greatest average classification accuracy. Model performance on the GTZAN dataset resulted in a 71.5% accuracy when classifying for genre. Model performance on the handpicked dataset resulted in an 83.3% accuracy when classifying for group labels.

There are many possible answers for the discrepancy in accuracy between the two datasets. One possible answer is that the much larger GTZAN dataset provides a more realistic representation of classification accuracy. Another answer may be that sound similarity in the hand picked dataset is much closer than the genres used in the GTZAN dataset. The lack of song duration as a possible feature for the GTZAN dataset may have also negatively impacted its accuracy.

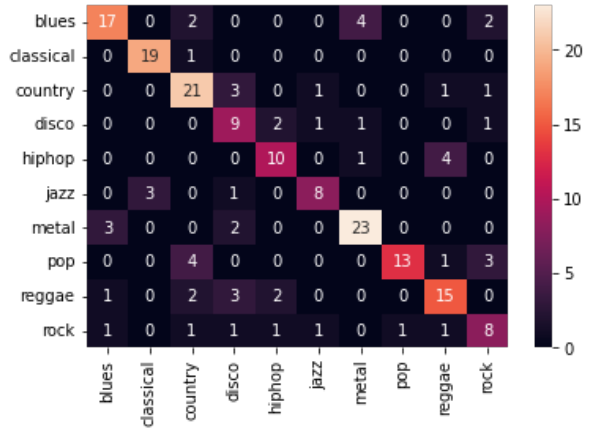

Fig 4: GTZAN results

A closer inspection of the GTZAN results shows that there is a massive difference in accuracy depending on the genre. The highest performing group, classical, correctly identified 19 out of 20 songs. The lowest performing group, rock, correctly identified only 8 out of 15 songs. The following results lead to the conclusion that rock may simply be a broad genre that contains too many elements of different genres and reinforces the belief that genre is a poor way to classify similar music.

On the other hand, the hand-picked dataset is too small to have a meaningful result without multiple tests. Multiple tests still resulted in an average accuracy of above 75%, and groups tended to perform around the same relative to other groups. The highest accuracy group, on average, belonged to group 1, which contained more extreme noisier songs.

8. CONCLUSION

The findings of the project reinforce the statement that genre isn't a particularly good way to separate music, as stated in the introduction. Current successful music recommendation systems such as last.fm highlight the fact that collaborative filtering can work well in a saturated system with a large community. Testing on the subject further shows that classification based majorly on MFCC features can expand this recommendation system to also work on lesser known songs.

There are several limitations that need to be addressed in this paper. The first and perhaps the main one was time. A strict 10-week structure left little time for any alterations or deviations from the initial proposal. Having labels created from collaborative filtering results was a decision made partway down the project that left little time to create a more organic way to create a dataset that was based off of collaborative filtering.

Another limitation was scope. The amount of different types of music are near infinite. While introductory findings on this scale are helpful as a proof of concept, they do not necessarily prepare for a fully functioning system.

Despite these limitations, the initial outlook on classification as a means to help recommendation systems seems positive.

9. FUTURE WORK

Limitations addressed in the conclusion will be worked on. The features and classification model themselves seem to work fairly well, but more

study will need to be done on organically providing labels to songs that can be used for classification purposes.

This assumes adding a limitation to the number of groups when generating them. Some form of clustering will be looked into to create an initial spread of labels that can be expanded upon. A very large (larger than GTZAN) dataset would be ideal to start this process.

10. REFERENCE

Milner, B. & Shao, Xu. (2006). Clean speech reconstruction from MFCC vectors and fundamental frequency using an integrated front-end. Speech Communication, 48(6). https://doi.org/10.1016/j.specom.2005.10.004

Singh, S. & Rajan, E. G. (2011). Vector Quantization Approach for Speaker Recognition using MFCC and Inverted MFCC. *International Journal of Computer Applications, 17*(1)http://dx.doi.org/10.5120/2188-2774

Wang, X. (2020). Research on Recognition and Classification of Folk Music Based on Feature Extraction Algorithm. *Informatica, 44*(4), 521-525. http://dx.doi.org/10.31449/inf.v44i4.3388

Loughran, R., Walker, J., O'Neill, M., O'Farrell, M. (2008). The Use of Mel-frequency Cepstral Coefficients in Musical Instrument Identification.

McFee, B., Raffel, C., Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. "librosa: Audio and music signal analysis in python." In Proceedings of the 14th python in science conference, pp. 18-25. 2015.

Aggarwal, C. C. (2015). Data classification : algorithms and applications (1st edition). CRC Press. https://doi.org/10.1201/b17320

Pandeya, Y. R., & Bhattarai, B. (2021). Deep-Learning-Based Multimodal Emotion Classification for Music Videos. *Sensors, 21*(14), 4927. http://dx.doi.org/10.3390/s21144927

Li, T., Ogihara, M., & Li, Q. (2003). A Comparative Study on Content-Based Music Genre Classification. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 282–289). Association for Computing Machinery.

Murthy, S. & Koolagudi, S. (2018). Content-Based Music Information Retrieval (CB-MIR) and Its Applications toward the Music Industry: A Review. ACM Comput. Surv. 51, 3, Article 45 (July 2018), 46 pages. https://doi.org/10.1145/3177849

Han, B., Rho, S., Jun, S., & Hwang, E. (2010). Music emotion classification and context-based music recommendation. *Multimedia Tools and Applications, 47*(3), 433-460. http://dx.doi.org/10.1007/s11042-009-0332-6

Vall, A., Dorfer, M., Eghbal-zadeh, H., Schedl, M., Burjorjee, K., & Widmer, G. (2019). Feature-combination hybrid recommender systems for automated music playlist continuation. *User Modeling and User - Adapted Interaction, 29*(2), 527-572. http://dx.doi.org/10.1007/s11257-018-9215-8

Lena, J. C., & Peterson, R. A. (2008). Classification as Culture: Types and Trajectories of Music Genres. *American Sociological Review, 73*(5), 697-718. https://www.proquest.com/scholarly-journals/classification-as-culture-types-trajectories/docview/218798564/se-2?accountid=1230

Miller, L. (2020, August 20). *So what does a music label do these days?* dot.LA. Retrieved October 10, 2021, from https://dot.la/new-music-tech-is-making-it-easier-and-harder-than-ever-for-artists-2647043299/so-what-does-a-music-label-do-these-days.

11. APPENDIX

Github Repository: https://github.com/brianrgeving/CS687_Capstone

Presentation: https://youtu.be/Sc0NHk7iONo

Demo: https://youtu.be/1wgF8Qa5pPk