Music Genre Classification using Neural Networks

Katon Minhas (kminhas@g.ucla.edu)
Department of Psychology, 502 Portola Plaza
Los Angeles, CA 90095 USA

Abstract

In this paper, I approach the task of categorizing music as one of ten different genres using a variety of deep learning models. In total, three different neural networks were constructed and trained using different input formats: A 3-Layer Dense Network taking a set of audio feature vectors, a 3-Layer Dense Network taking a set of pixel-value-derived vectors as input, and a 3-Layer Convolutional Neural Network (CNN) taking 2D spectrogram images as input.

The results on the GTZAN dataset show a 0.334 accuracy rate for the Feature-Analysis Dense model and a 0.322 accuracy rate for the Spectrogram-Based CNN model. The Spectrogram-Based Dense model achieved a 0.205 accuracy rate. To further test the model, I explored its functionality against 25%, 50%, and 75% noise introduced to the spectrogram data. The Feature-Analysis model held up best to noise, achieving 0.209 accuracy at 25%. The Spectrogram-Based Dense model performed with 0.146 accuracy, while the Spectrogram-Based CNN model did not perform better than chance.

These results were consistent with predictions that Feature Analysis would perform the best, especially against noise, due to it being the closest approximation to human hearing.

Code, additional figures, and data can be found in the project repository at https://github.com/katonminhas/MusicGenreClassifier

Keywords: convolutional neural network, feature extraction, spectrogram

Introduction

With the increasing popularity and interest in music, as well as accessible platforms, there is more of a need to categorize and organize songs. Genre categorization is a way to organize music, while also being beneficial for discovering trends in genre and artist popularity. Streaming platforms can also provide users with song suggestions that fit their genre preference. As the size of a music database increases, it becomes more difficult to manually complete such a task.

Previous literature in the area of music genre classification has focused on the extraction of meaningful features in an effort to simulate the process of human hearing. By selectively including the audio features which create clear distinctions between genres, a network can exclude irrelevant information and greatly improve accuracy. However, selective inclusion of features may also discard relevant information, as the entire audio file is never fully examined as a whole. To account for this problem, analysis has also been conducted using spectrograms, which serve as visual representations of the full-form audio file. This has the benefit of including all available information. However, it also includes potentially irrelevant and ambiguating details.

The purpose of this project was to determine whether a feature-based or a spectrogram-based analysis would perform better in a music-genre classification task. I also sought to determine which of the methods would be most robust when presented with noisy data. To answer these questions, 3 separate neural networks were developed: A 3-Layer Feature-Based Dense Network, a 3-Layer Spectrogram-Based Dense Network, and a 3-Layer Spectrogram-Based Convolutional Network.

It was hypothesized that feature analysis would perform best in both normal and noisy contexts. This prediction is based on the higher selectivity inherent in a feature-based model. When a person makes a judgement as to the genre of a piece of music, they factor in input from a variety of auditory features and select which are most relevant to their decision based on previous knowledge and experience. Analyzing various physical auditory features in conjunction with each other is very similar to the way humans analyze sound in the real world. Thus, feature analysis should provide the best imitation of actual human performance. By contrast, spectrogram analysis uses the equivalent of visual differentiation (by taking each pixel value of a spectrogram at face value). While this is similar to how humans analyze and differentiate visual images, it is likely not the best method for classifying audio data. Simply looking at the spectrograms of different genres will reveal that it is a much more difficult task for people to visually differentiate spectrograms than auditorily differentiate a short audio clip. In addition, taking every pixel of a spectrogram image gives equal weight to both relevant and irrelevant information, which would likely decrease classification performance.

Literature Review

Audio signals have many features that are visualized through spectrograms. Looking at the characteristics that are most relevant to classifying genres by extracting features will assist in analysis. In his research, Dan Ellis explains how chroma features can help analysis through representation of 12 distinct semitones in the musical octave. Variation and progression of semitones throughout a piece of music is considered an important distinguishing feature of musical genres. Another helpful feature is spectral contrast, which Jiang et al. found to have better performance in distinguishing genres than Mel-Frequency Cepstral Coefficients (MFCC) in their research. While MFCC represents only the average of the spectrum, spectral contrast represents relative spectral characteristics by considering peaks and valleys separately to display the distribution. Spectral flatness is an additional feature that is useful in contributing to feature analysis. In Notes on Measures for Spectral Flatness, Nilesh Madhu justifies the usefulness of this feature, citing its ability to call attention to more prominent wave features.

Previous networks have focused on one or more audio features. Haggblade et al devised a network focusing on MFCC to classify between 4 genres: metal, jazz, classical, and pop. The project used multiple models, including a single hidden-layer neural network with a softmax activation function, to classify songs by genre. Other feature-based networks have focused on features such as the spectral centroid and chroma frequencies, with varying degrees of success. There has been limited focus on spectrogram-only networks.

Method

Dataset

I used music samples from the online GTZAN database (the full database can be found at http://marsyas.info/downloads/datasets.html). The dataset consists of 1000 songs, 10 genres, and 100 songs per genre. The database was compiled from 2000-2001 “from a variety of sources including personal CDs, radio, and microphone recordings”. Genres in the dataset are pre-identified as one of the following ten genres: Blues, Classical, Country, Disco, Hip-hop, Jazz, Metal, Pop, Reggae, and Rock. Each song clip from the database was originally 30 seconds long. In order to preserve memory and decrease processing time, random 5-second samples of the original clips were taken to be used for analysis.

Spectrogram Analysis

A spectrogram is a time-frequency 2D representation of an audio signal that plots amplitude (color) and frequency (y-axis) of an audio signal against time (x-axis). Using a fast-fourier transform algorithm (provided by the librosa library), the 5-second audio clips were converted into spectrogram images, resembling the one pictured in figure 1. The pixel values were then multiplied by a scalar to increase contrast. For the Convolutional Neural Network, the entire 128x216 spectrogram of each song was taken as input to train the network.

For the Spectrogram-Based Dense Network, each spectrogram was condensed into a 128x1 vector, with each vector value consisting of the mean pixel value of its corresponding row in the complete spectrogram. This method preserves the general color spectrum of the image while sacrificing local details, as well as the temporal dimension of the song. The purpose of conducting analysis on a 1D version of the spectrograms was to decrease the computational time and storage space needed for increasingly large datasets. In datasets larger than 1000 songs, inputting an entire set of 2D spectrogram images could prove too computationally expensive to be feasible.

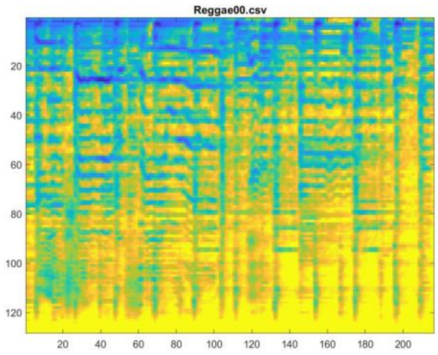

Figure 1: The spectrogram image of reggae00.wav. Each of the 1000 5-sec clips were converted into 128x216 images like this one.

Feature Analysis

Audio signals are made up of many features that can be extracted and converted into meaningful values for classification. Feature extraction was done using librosa - a python package for music and audio analysis. The goal of the feature extraction process was to carefully select the features of an audio signal that are most relevant for genre identification. The following seven features were identified as the features that people most likely focus on when listening to music:

Beats Per Minute (BPM) BPM refers to the tempo, or speed, of a music sample. While BPM can be an important distinguishing feature of genres, the distribution of BPMs within each genre is often very wide. This makes BPM a somewhat unreliable feature when taken without additional feature information.

Zero Crossing Rate (ZCR) Zero crossing rate refers to the rate at which a sign changes from positive to negative along an audio signal. It is often used in speech recognition and music information retrieval. In general, higher values of ZCR are seen in genres that place more emphasis on sharp percussive sounds, such as rock or hip hop, while lower values are seen in less percussive genres such as classical or jazz.

Spectral Centroid The spectral centroid is a representation of the center of mass of the spectrum. This value is calculated using means of frequencies weighted according to their magnitude. If the magnitudes of high and low frequencies in a song are fairly balanced, the spectral centroid would be near the center of the spectrum. The spectral centroid is often used as a representation of timbre, with brighter sounds (sounds with wide high-end features such as a trumpet or synthesizer) having a higher centroid than less bright sounds.

Spectral Flatness Otherwise referred to as the tonality coefficient, spectral flatness quantifies how tone-like a sound is (quantified by the number of local minimums and maximums as defined by a given magnitude), in contrast to being noise-like (flat sound similar to white-noise). It is calculated by taking the ratio of the geometric mean to the arithmetic mean from the magnitude of an audio signal spectrum (Madu, 2009). A high spectral flatness value, which is closer to 1.0, represents a spectrum that is similar to white noise.

Chroma Frequencies The chroma feature vector represents the entire spectrum of the audio within 12 bins corresponding to the 12 chroma of an octave. (Ellis, 2007). The value stored in the vector is indicative of the octave of a pitch. Chroma frequency analysis is beneficial for high-level semantic analysis, such as harmonic and melodic characteristics of music and chord recognition.

Spectral Contrast Spectral contrast compares the mean energy in the top quantile (spectral peak) to the bottom quantile (spectral valley) from each from of a spectrogram. This calculated spectral difference is stored within each frequency band (Jiang et al., 2002).

Mel Frequency Cepstral Coefficients (MFCC) MFCCs are useful for tasks such as speech recognition because the are features that compactly represent the speech amplitude spectrum. MFCCs, when plotted against the mel scale, model the human perception of changes in pitch, as well as the tonal characteristics of a human voice.

The above features were extracted from the clips and stored in a set of 1000 26x1 vectors, with each vector value corresponding to one of the audio features (MFCC contributed 20 values for indexes 7-26). In addition, a 27th index was added to act as a tag (with values 0-9) indicating the correct genre. Each feature value was then fit between 0 and 1 using a min-max fit algorithm to maintain distribution across songs. Finally, each song vector was normalized.

Training and Testing

To analyze our data, 3 different networks were created. Each network was first trained on a set of 800 songs, selected randomly and evenly distributed such that 80 songs came from each genre. The remaining 20 songs/genre were used for testing. Each training phase was standardized to be run for 10 epochs with a batch size of 128. The networks were constructed as follows:

3-Layer Feature-Based Dense Neural Network

This network used 3 densely-connected layers of 64, 32, and 10 neurons each (10 corresponding to a prediction of the input song's genre). The network took as input a series of 26 dimensional feature vectors, as specified in the previous section. The first two layers used a ReLU activation function, while the final output layer was constructed using a SoftMax activation. During development of the network, various activation functions were experimented with to achieve similar results.

3-Layer Spectrogram-Vector-Based Dense Neural Network

This network only differed from the prior in the number of neurons in the first two layers. The densely connected layers have 128, 64, and 10 layers each. The dimensions of the layers were chosen to correspond to the increase in dimensionality of the input vectors to 128.

3-Layer Spectrogram-Based Convolutional Neural Network

The CNN takes as input the collection of spectrograms as a series 128x216 matrices. The 3 layers had dimensionalities of 64, 32, and 10 respectively. Like the previous models, ReLU activation was used for the first two layers, while SoftMax activation was used for the third. Since the spectrograms appeared to differ greatly on a very local level rather than on a broader scale, a relatively small kernel size of 3 was chosen for the CNN. Experimenting with larger kernel sizes tended to produce poorer results.

In addition to standard, uncorrupted input, the networks were tested on their ability to withstand noise. To achieve this, a percentage of pixels were randomly selected to have their values set to 0. This was done with 25%, 50%, and 75% of pixels in the spectrogram images. Once the noisy data was created, each network was trained using the full set of 1000 uncorrupted songs, then tested using the set of 1000 noisy songs to determine network performance against noise.

Results

Table 1: Summary of Network Performance

	Feature	Spec (Dense)	Spec (CNN)
No Noise	0.3341	0.2050	0.3215
25% Noise	0.209	0.1460	0.1050
50% Noise	0.1460	0.1000	N/A
75% Noise	0.1000	0.1000	N/A

Table 1: Test set accuracy results are shown. Accuracy of 0.1 represents chance. CNN tests at 50% and 75% noise were not conducted.

Consistent with predictions, our Feature-based Dense Network as well as the Spectrogram-Based CNN performed best in the non-noisy conditions. Although the Spectrogram-Based Dense Network performed at roughly twice of chance accuracy, it was unable to achieve the levels of success experienced by the other two networks.

When noise was introduced, the Spectrogram-Based CNN experienced the most drastic decrease in performance, dropping to chance levels when presented with just 25% noise (trials were not run at 50% and 75% for this reason). By contrast, both Dense Networks held up reasonably well to noise, performing significantly higher than chance, with the Feature-Based network still achieving the best performance. At 50% noise levels, both Spectrogram-Based networks experienced (or were assumed to experience) chance performance, while the Feature-Based network performed slightly better, though still achieving just 14.6% accuracy. At 75% noise levels, none of the networks performed better than chance.

A full report of network results, including training set performance and additional graphs, can be found in the project repository.

Discussion and Future Implications

Our results ultimately confirmed our hypothesis that a feature-based system would be best for discriminating different genres of music. While both the Feature-Based Dense Network and the Spectrogram-Based CNN performed equally well with no noise, the Feature-Based network was able to identify genres better than chance with some noise added. By contrast, performance in both Spectrogram-Based networks decreased significantly when noise was added.

The high performance of the feature-based network can be attributed to our careful selection of features to include. I attempted to include only the auditory features which create clear distinctions between genres. This was done in an effort to simulate the human process of distinguishing music genres. When a person listens to a piece of music, they classify it based on familiar or unfamiliar features rather than taking into account the song as a whole. Features such as BPM and chroma frequencies are naturally detected by human ears and used to make judgments about the song being heard. By selecting only the relevant features, a large amount of irrelevant information was eliminated and classification process was focused on only the most pertinent aspects of the song.

The high performance of the Spectrogram-Based CNN is likely due to the nature of the network itself. Convolutional neural networks have long been considered a reliable method of image recognition. By choosing a relatively small kernel size of 3, it was ensured that the CNN would pick up appropriately local details needed to distinguish genre features.

The relatively low performance of the Spectrogram-Based Dense Network can be attributed to the low fidelity of the input. By only taking the mean row values of the image, each song was condensed past the point of reliable recognition. While this does have the benefit of eliminating time factors (songs are the same forward and backward, across different samples etc), it does decrease performance significantly.

With regard to additional noise, the relatively high performance of the Feature-Based network can be attributed to the way in which the noise was added. By evenly distributing the zeroed-out pixel values, most features were left very slightly changed. For example, a feature like the spectral centroid (which attempts to locate a ‘center of mass’ of the spectrogram) would likely be virtually unchanged by evenly distributed noise. By contrast, the Spectrogram-Based CNN relies on picking up fairly local features from the spectrogram, which would become highly distorted with even a slight amount of noise. The Spectrogram-Based Dense Network performed relatively well under noise, again due to the nature of the input. When a small amount of noise is added, the mean values of each row will decrease only slightly, allowing the network to perform similar to its performance with no noise. The network only becomes unreliable when a larger portion of the spectrogram image is corrupted.

One major limitation of our study concerned the scope of the music samples used in the training and testing sets. The GTZAN database was compiled before 2002, making it incapable of adapting to the natural evolution of musical genres into the modern era. Furthermore, there was not much diversity within genres in the database. Each sample seemed to be a very representative example of its genre, with few major outliers included. This is not reflective of the spectrum-like nature of musical genres. In reality, songs frequently straddle the line between two or more genres, which can often be detected by experienced listeners when exposed to even a 5 second snippet.

To address these issues in future research, clips obtained from modern music should be incorporated into the set. This should serve to help the model gain a better idea of what is “characteristic” of a given genre. In addition, songs that do not clearly fit into a given genre should be included in training, potentially with multiple correct tags to indicate the variety of genres that could count as correct. This will enable the network to identify at least one correct genre of multi-genre songs.

Acknowledgments

I would like to acknowledge assistance from Dr. Zili Liu as well as Lucy Cui.

References

Dan Ellis. 2007. Chroma feature analysis and synthesis. *Resources of Laboratory for the Recognition and Organization of Speech and Audio-LabROSA.*

Jiang, Dan-Ning & Lu, Lie & Tao, Jian-Hua & Cai, Lian-Hong. (2002). *Music type* *classification by spectral contrast feature.* 113 - 116 vol.1. 10.1109/ICME.2002.1035731.

Madhu, Nilesh. (2009). Note on measures for spectral flatness. *Electronics Letters.* 45. 1195 - 1196. 10.1049/el.2009.1977.

Tzanetakis, George. (2002). GTZAN Genre Collection. *Marsyas (Music Analysis, Retrieval and Synthesis for Audio Signals).* http://marsyas.info/downloads/datasets.html.