CS543: Music Genre Recognition through Audio Samples

Vedant Choudhary
Rutgers University
New Brunswick, New Jersey 08901
vedant.choudhary@rutgers.edu

Aditya Vyas
Rutgers University
New Brunswick, New Jersey 08901
aditya.vyas@rutgers.edu

Abstract

This project is part of our coursework for CS543. We aim to implement music genre recognition through audio samples using deep learning techniques. The idea of the project is that audio samples provide advanced features which are more accurate at recognizing a genre than using lyrics. Audio samples can be converted to image spectrogram, reducing the problem to that of an image classification. This allows Convolutional Neural Networks to find features of the audio sample. The data set considered is Free Music Archive (FMA) data set which consists of 106,574 tracks, 16,916 artists, 15,234 albums, 161 genres.

1 Introduction

Recognizing genre of a music through lyrics has been a well researched area, but is still a naive approach. It suffers from the limitation that it does not use any musical feature of the song, like the frequency of tones or pattern of beats. In our project, we are trying to overcome this limitation by incorporating audio samples and deriving features from it to present a better, more accurate genre representation of a song. We go along the project by using the ***small*** dataset of FMA, which has 8,000 tracks of 30s, and 8 balanced genres.

1.1 Problem Statement

Music Information Retrieval (MIR) is a wide research area. It encompasses fields like machine learning, signal processing and musical theory. Music Genre Recognition (MGR), the topic for our project is one of the most widely tackled problem in MIR. According to (Costa et al., 2011), a genre of a song is a categorical variable created by humans to facilitate easy grouping of songs into their respective style of music. With the advent and boom of internet services, music streaming applications are coming up rapidly and it is no longer feasible to manually tag a song with its genre. Further, there are so many sub-genres that it is difficult to manually assign a list of genres to songs. This is how music genre recognition softwares are aiding such streaming applications to maintain a proper meta data of songs in their databases. MGR further helps categorize songs based on genres, which can be used as a music collection for users to listen to. Additionally, in applications such as recommending music, or generating music, genre is used as an important feature, further emphasizing the importance of correctly recognizing genres.

2 Literature Review

Music information retrieval (MIR) has gone through a lot of research like (Bertin-Mahieux et al., 2011), where the authors introduced a dataset containing millions of songs along with its audio features and metadata. The work gave rise to a lot of problems in MIR, such as (Bertin-Mahieux and Ellis, 2011) - recognizes cover songs, (Mishra et al., 2016) - song year prediction in Apache Spark, among others.

There has been work in recognizing genre through lyrics, like in the paper by (Tsaptsinos, 2017). According to our research, one of the oldest papers we could find on music genre is (Tzanetakis and Cook, 2002), which categorized music genre recognition task as a pattern recognition task. In the paper by (Dieleman et al., 2011), they used a pre-trained convolutional neural networks on audio features like timbre and chroma for genre recognition, artist recognition and key detection.

Music genre recognition task can also be tackled using spectrograms for genre recognition (Costa et al., 2013), in which they have extracted features from spectrograms using local binary patterns, and then predicted by using an ensemble of classifiers. They have used Latin Music Database and are able to attain 83% recognition rate. Additionally, in (Grzywczak and Gwardys, 2014), the researchers have used CNNs on spectrograms after pre-training it on ILSVRC-2012, and were able to achieve an accuracy of 70% on the GTZAN dataset. In our project, we intend to recognize a genre with time on FMA dataset, which is a fairly new music dataset.

3 Data

After exploring the internet for different possible open source datasets, we finalize **FMA** dataset as the final dataset for the project. There are other possible datasets related to music available on net, but lack some of the qualities which we need for our problem. Shown below are the possible datasets and their limitations:

3.1 Million Songs Dataset

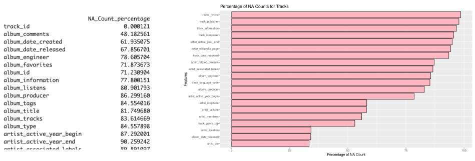
Million songs dataset is one of the largest music datasets available on net, however, it does not provide audio clips which are of utmost importance to us. It asks the user to download music clips from online streaming sites like 7digital, which just adds on to the complexity of the problem. Further, it's dataset does not allow to recognize genre, and mostly focuses on topics like segmentation, automatic tagging, year recognition, lyrics etc.

3.2 GTZAN Dataset

It is one of the widely used datasets available for music genre recognition available on net. However, it only has 1000 audio samples for 10 genres i.e. 100 samples per genre. Further, it has lot of faults like mislabeling, repetitions, and distortions (Defferrard et al., 2016). Then, the data was collected in 2001-2002, a time when there was not much electronic music available. Right now, electronic music is one of the most listened genre, so it's inclusion in analysis is of high importance. Lastly, it misses metadata. More information about it's cons can be found in (Sturm, 2013).

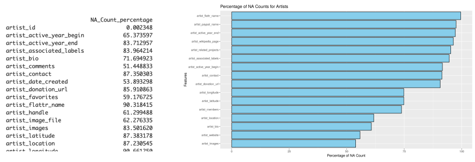
3.3 Free Music Archive - FMA

The data set used for our problem is FMA dataset. It is a huge dataset of audio samples and metadata, consisting of 917 GB of licensed audio from 106,574 tracks and 161 genres (Defferrard et al., 2016). However, due to processing power limitations, we will analyze and solve the problem on *small* dataset, which is like GTZAN data, containing 8,000 tracks of 30s, and 8 balanced genres - electronic, experimental, folk, hiphop, instrumental, international, pop, rock.



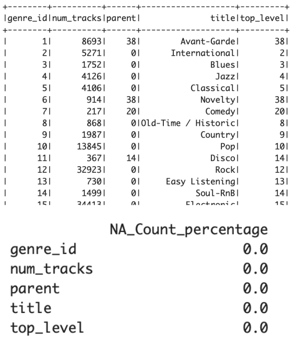
```
                           NA_Count_percentage
track_id                              0.000121
album_comments                       48.182561
album_date_created                   61.935075
album_date_released                  67.856701
album_engineer                       78.605704
album_favorites                      71.873673
album_id                             71.230904
album_information                    77.800151
album_listens                        80.901793
album_producer                       86.299160
album_tags                           84.554016
album_title                          81.749680
album_tracks                         83.614669
album_type                           84.557898
artist_active_year_begin             87.292001
artist_active_year_end               90.259242
artist_associated_labels             89.891097
```

Figure 1: Missing values in Tracks data



```
                           NA_Count_percentage
artist_id                             0.002348
artist_active_year_begin             65.373597
artist_active_year_end               83.712957
artist_associated_labels             83.964214
artist_bio                           71.694923
artist_comments                      51.448833
artist_contact                       87.350303
artist_date_created                  53.893298
artist_donation_url                  85.910863
artist_favorites                     59.176725
artist_flattr_name                   90.318415
artist_handle                        61.299488
artist_image_file                    62.276335
artist_images                        83.501620
artist_latitude                      87.383178
artist_location                      87.230545
artist_longitude                     90.661250
```

Figure 2: Missing values in Artists data



```
+--------+----------+------+-------------------+---------+
|genre_id|num_tracks|parent|              title|top_level|
+--------+----------+------+-------------------+---------+
|       1|      8693|    38|        Avant-Garde|       38|
|       2|      5271|     0|      International|        2|
|       3|      1752|     0|              Blues|        3|
|       4|      4126|     0|               Jazz|        4|
|       5|      4106|     0|          Classical|        5|
|       6|       914|    38|            Novelty|       38|
|       7|       217|    20|             Comedy|       20|
|       8|       868|     0|Old-Time / Historic|        8|
|       9|      1987|     0|            Country|        9|
|      10|     13845|     0|                Pop|       10|
|      11|       367|    14|              Disco|       14|
|      12|     32923|     0|               Rock|       12|
|      13|       730|     0|     Easy Listening|       13|
|      14|      1499|     0|           Soul-RnB|       14|
|      15|     34413|     0|         Electronic|       15|

            NA_Count_percentage
genre_id                    0.0
num_tracks                  0.0
parent                      0.0
title                       0.0
top_level                   0.0
```

Figure 3: Genre data visualization

dataset	tracks	genres	length	size
small	8,000	8	30	7.4
medium	25,000	16	30	23
large	106,574	161	30	98
full	106,574	161	278	917

Table 1: Subsets of FMA

3.3.1 Exploratory data analysis on FMA

In this section, we present why the dataset chosen for the project is a feasible one and explain more about its meta data and correctness.

There are 3 major datasets – *genre.csv, artists.csv, tracks.csv* – which need to be explored to get a better understanding of the FMA dataset and how it can be used for our project.

We first look at snippets of data files and find out how many missing values are present in different columns. It is an important step to establish the authenticity of data since a large number of missing values in certain columns can hinder the project modelling phase. Ex- for our project, it is of utmost importance that there should be minimal missing data in columns such as *"track_id", "genre_id", "artist_id", "album_id" etc.* for us to be able to go ahead with the project. Although, Figures 1 and 2 have a lot of columns with missing value. It is of no concern to us, because these columns do not convey any meaning towards genre recognition. Having missing values in "artist_bio" does not impact our analysis and project.

On exploring *genre.csv*, we find out that there are no null values in the data 3. This finding validates our decision to go forward with FMA because genre data is of utmost importance. Fig 4 shows how tracks are diversified under genres. Due to image readability, we show track counts only for top 20 genres. It further shows that the dataset is in sync with current musical trends. As stated above, electronic music is one of the most listened genre in today's world and FMA dataset correctly takes that into account. This inclusion of electronic music is something which lacks in GTZAN dataset.

Fig. 5 shows the count of sub-genres in parent genres and Fig. 6 shows the variety of genre among the tracks of top artists. Both these figures highlight the detailed level-data available for genre recognition tasks. Additionally, Fig. 7 displays the top 20 artists present in the dataset. It can be noticed that the figure validates the fact that there is no biased artist variable in the dataset.
With this thorough analysis of data, it is evident that FMA is a good option for conducting MGR.

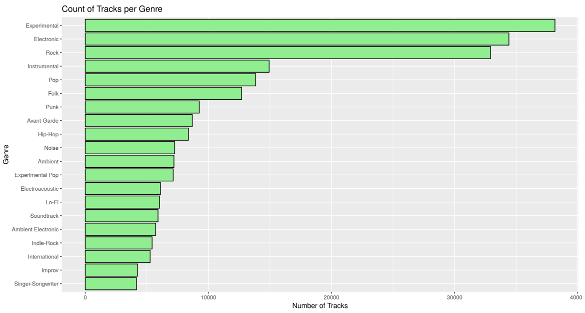


Figure 4: Tracks per genre

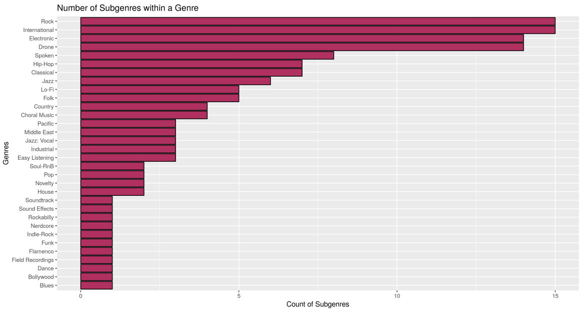


Figure 5: No. of sub-genres of parent genres

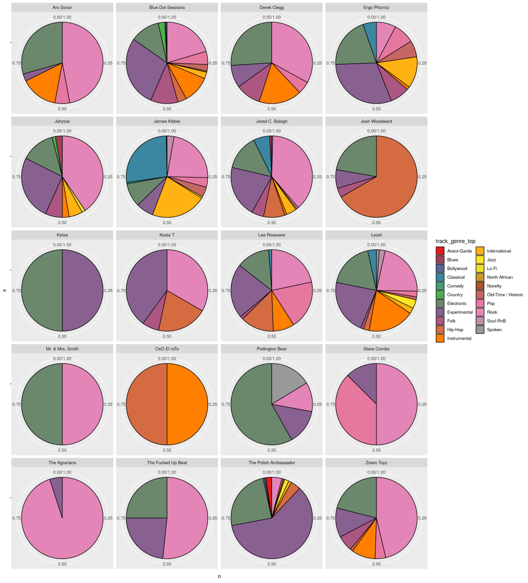


Figure 6: Top Genres of Top Artists

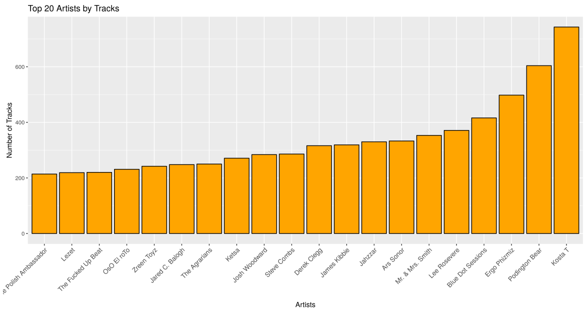


Figure 7: Top 20 Artists

4 Proposed Approach

Till now, we have discussed our idea of using audio samples to achieve music genre recognition capability without touching upon how to achieve that. In this section, we will try to go through the solution we are proposing.

Lyrics-based genre recognition achieves subpar performance and hence, we aim to achieve higher accuracy by using audio features, primarily spectrograms of a track.

Spectrograms are image representation of the spectrum of frequencies in a track as it varies with time. In the final paper, we will explain this concept in much more detail. For now, we want to establish the fact that we aim to harness this image representation of a track and use them as an input for our convolutional neural network.

Convolutional neural networks(CNNs) are considered the paragon of any image processing technique used nowadays because of it's ability to learn image features accurately and therefore, we aim to harness the power of CNNs on our input of spectrograms. The architecture of CNN is yet to discuss, but we will try out different architectures ranging from basic ones to more complex ones.

Our main aim is to not just generate a single genre per song, but generate a continuous genre prediction through the song. This stems from the nature of songs: although they are categorized to some parent genre, it contains elements of a lot of different genres. We believe a continuous prediction of genre for the song will do justice to the input spectrograms (as they provide frequency plots per second). As of now, we believe an LSTM (Long short-term memory) model can be applied for this sequential data processing.

We hope to present the final outcome in the form of a visualized web page or application, where the audio sample is played and the model predicts genre through time.

5 Approach

First, let us talk about spectrograms and their importance for the task at hand. Spectrograms capture the frequency spectrum of a song with time, which is just what we need to predict the genre belief of a song with time. While going through internet, we got to know through (Dieleman, 2014) that instead of using raw spectrograms, it is better to use mel-spectrograms (transforming frequencies to mel-scale), which is the scale in which humans perceive different frequencies. A mel-spectrogram can be analysed in the similar way a spectrogram is analysed. The points which have high peaks are notes played over a short time, while chords are seen as long stretches of horizantal line since they are played over time (shown in Fig. 8)

After extracting images through spectrograms from FMA dataset, we trained the images on an end-to-end Convolutional Recurrent Neural Network model with 70/30 training/validation ratio. The architecture for the model has been inspired from (Shi et al., 2017) and (Kozakowski and Michalak, 2016). Initially, the images are gone through 3 convolutional layers to extract the features from the spectrogram. After each convolutional layer, the model is passed through ReLU activation and max pooling. We have used Convolution in 1-D and not 2-D as we are interested in finding out patterns with respect to time.

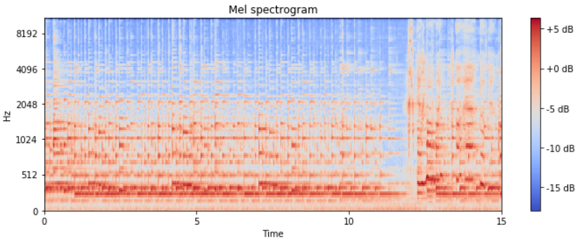
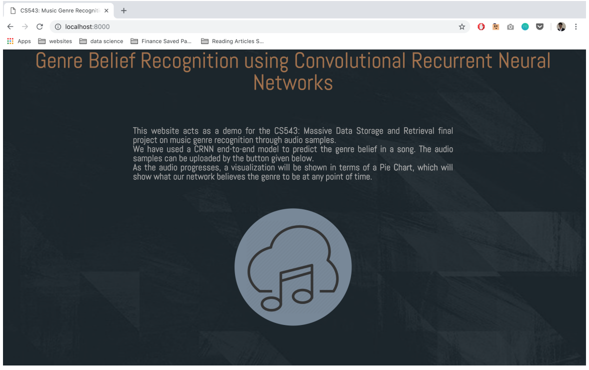


Figure 8: Mel Spectogram of an audio clip



Genre Belief Recognition using Convolutional Recurrent Neural Networks

This website acts as a demo for the CS543: Massive Data Storage and Retrieval final project on music genre recognition through audio samples.
We have used a CRNN end-to-end model to predict the genre belief in a song. The audio samples can be uploaded by the button given below.
As the audio progresses, a visualization will be shown in terms of a Pie Chart, which will show what our network believes the genre to be at any point of time.

Figure 9: Web application for the project

After getting the features, a time distributed layer has been used, which gives out probabilities for the 8 genres we intend to recognize with every timestep. Since our genres are one-hot encoded, we use a categorical cross-entropy loss metric along with Adam optimizer with an initial learning rate of 0.001. After running this model on different parameters, the best we can get is a training accuracy of 85% and validation accuracy of 60%. We agree that 60% sounds not that good, but according to our knowledge, we do not see many people using this dataset for music genre recognition, and even the ones who have done (Hebber, 2017), have not tried recognize genre with time.

We also tried different architectures, however, the one mentioned here, gave us a good accuracy in both training and validation, without overfitting.

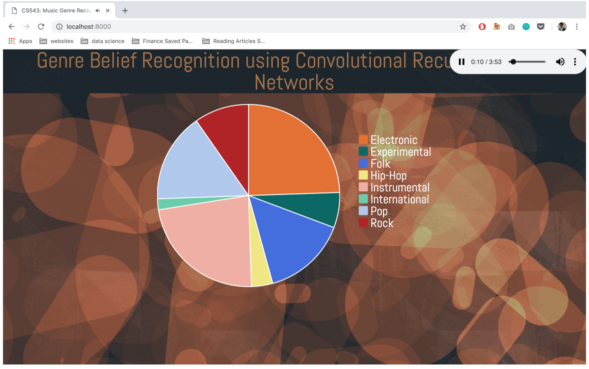


Figure 10: Visual output

Architecture	T. Accuracy	V. Accuracy
TD	85%	60%
LSTM	60%	52%
Spotify	97%	55%

Table 2: Architectures

TD stands for Time Distributed model explained in detail. LSTM model used an LSTM layer before Time Distributed layer, keeping everything same, however, we found that LSTM model was too slow to learn and could not surpass TD. Spotify model is the model we adopted through (Dieleman, 2014). It gave us a high training accuracy and low validation accuracy, which means it overfits our data. Also, according to analysis provided by the authors in (Defferrard et al., 2016), their baseline genre recognition model only achieved 12.5%.

We tried hyper-parameter tuning in the given models too, however, no change seemed to impact drastically.

6 Conclusion and Future Work

In this project, we were able to solve the problem of recognizing the genre of a song. However, unlike most of the work done in this field, which is giving out a single genre for the whole song, we tried to express the belief of a song with time. We were successful in achieving that, although with a 60 % accuracy. We believe that representing genre probabilities with time also helps us understand how the model is behaving/what it understands, which we can observe visually through the web application created. Moreover, recognizing a genre is an important task for music streaming companies as it allows them to categorize songs based on genres, use it for recommendation and creating customized playlists.

In future, we will try to improve upon the model as right now, it seems there is a bias in hip-hop genre. We believe we can use the work done by (Aytar et al., 2016) along with LSTM and Time Distributed layers to improve the model accuracy.

As a last note, upon the suggestion by professor, we tried to go through the research in the field of music generation. However, due to shortage of time, we could not implement it. But according to the intial research, we found some papers which can be beneficial to us. Most papers or projects have used MIDI dataset. We found (Mehri et al., 2016) and (van den Oord et al., 2016) of importance to us, which we plan to read and incorporate in future.

Acknowledgments

PySpark, *R* and *Python* have been used to run all the exploratory data analysis on this project. The libraries used are *seaborn*, *ggplot2*, *matplotlib* and *pandas*.

For the modelling part, we have used libraries such as *librosa*, *tensorflow*, *keras*, *numpy*, *pickle*, *tensorflowjs*. Since the computation power of our laptops are low, we have used *Google Colab* to train the model and convert it into JSON for web serving.

The web layout has been created by using *HTML*, *CSS*, *Javascript* using some *javascript* packages like *audio-utils*, *Chart.js*, *audio visualization tool*, *resampler.js*, siriwave.js.

The document has been made on *ACL2018* template on *Overleaf*.

References

Yusuf Aytar, Carl Vondrick, and Antonio Torralba. 2016. Soundnet: Learning sound representations from unlabeled video. In *Advances in Neural Information Processing Systems*, pages 892–900.

T. Bertin-Mahieux and D. P. W. Ellis. 2011. Large-scale cover song recognition using hashed chroma landmarks. In *2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 117–120.

Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. 2011. The million song dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*.

Y. M. G. Costa, L. S. Oliveira, A. L. Koericb, and F. Gouyon. 2011. Music genre recognition using spectrograms. In *2011 18th International Conference on Systems, Signals and Image Processing*, pages 1–4.

Yandre Costa, Luiz Soares de Oliveira, Alessandro Koerich, and Fabien Gouyon. 2013. Music genre recognition based on visual features with dynamic ensemble of classifiers selection. pages 55–58.

Michal Defferrard, Kirell Benzi, Pierre Vandergheynst, and Xavier Bresson. 2016. Fma: A dataset for music analysis.

Sander Dieleman. 2014. Recommending music on spotify with deep learning.

Sander Dieleman, Philémon Brakel, and Benjamin Schrauwen. 2011. Audio-based music classification with a pretrained convolutional network. In *Proceedings of the 12th international society for music information retrieval conference : Proc. ISMIR 2011*, pages 669–674. University of Miami.

Daniel Grzywczak and Grzegorz Gwardys. 2014. Deep image features in music information retrieval. volume 60, pages 187–199.

Rajat Hebber. 2017. music2vec: Generating vector embeddings for genre-classification task.

Piotr Kozakowski and Bartosz Michalak. 2016. Music genre recognition.

Soroush Mehri, Kundan Kumar, Ishaan Gulrajani, Rithesh Kumar, Shubham Jain, Jose Sotelo, Aaron Courville, and Yoshua Bengio. 2016. Samplernn: An unconditional end-to-end neural audio generation model. *arXiv e-prints*, abs/1612.07837.

P. Mishra, R. Garg, A. Kumar, A. Gupta, and P. Kumar. 2016. Song year prediction using apache spark. In *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1590–1594.

Aron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alexander Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. 2016. Wavenet: A generative model for raw audio. In *Arxiv*.

Baoguang Shi, Xiang Bai, and Cong Yao. 2017. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(11):2298–2304.

Bob Sturm. 2013. The gtzan dataset: Its contents, its faults, their effects on evaluation, and its future use.

Alexandros Tsaptsinos. 2017. Lyrics-based music genre classification using a hierarchical attention network.

G. Tzanetakis and P. Cook. 2002. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302.