INDIAN INSTITUTE OF TECHNOLOGY, BOMBAY

Music Genre Classification

Suraj Kumar, 193050003
Yadnyesh Patil, 193050067
Rajershi Gupta, 193059002
Aman Kumar Singh, 193050022

November 25, 2019

ABSTRACT

In this project our main aim is to analyse different classification models for music genres and attempt to improve their accuracy. We used the GTZAN genre dataset. Intensive pre-processing and and appropriate feature selection really helped us in our task. The features were mostly obtained from Chroma frequency, spectral centroid, spectral rolloff, zero crossing rate and MFCC vector.

Key words: classification models, music genres, GTZAN genre dataset, improve accuracy, feature selection, preprocessing

CONTENTS

1 INTRODUCTION

A music genre is a conventional category that identifies some pieces of music as belonging to a shared tradition or set of conventions. Music can be divided into different genres in many different ways. The artistic nature of music means that these classifications are often subjective and controversial, and some genres may overlap. Genres don't always depend upon the sound of music or the rhythm Sometimes they depend on history or geography of the place. For example the music of southern states of USA is called country music.

The dataset that we are working on contains wav (Waveform Audio File) format music files. We are using classifiers that classifies the music into 10 genres like Jazz, rock etc. This dataset was created by Tzanetakis and Cook and is considered as the standard dataset to be used for genre classification.

2 MAIN GOAL

In this project our main goal is to improve the accuracy of previously build genre classifiers, which have used the GTZAN dataset.

3 RELATED LITERATURE

A lot of work has been done in the field of music genre classification. The GTZAN dataset was created by Tzanetakis and Cook[1] who themselves done a lot of work in classification of music genre using machine learning techniques.

Companies nowadays use music classification, either to be able to place recommendations to their customers (such as Spotify, Soundcloud) or simply as a product (for example Shazam). Determining music genres is the first step in that direction. Machine Learning techniques have proved to be quite successful in extracting trends and patterns from the large pool of data[2]. The same principles are applied in Music Analysis also.

The paper[1] has achieved an accuracy of 61% .

4 Description of Approach

4.1 A brief introduction to dataset

We have used the GTZAN dataset.The dataset consists of 1000 audio tracks each 30 seconds long. It contains 10 genres, each represented by 100 tracks. The tracks are all 22050Hz Mono 16-bit audio files in .wav format.

4.2 Song features

The commonly used features for musical genre classification are:

1. **Timbral Features:** This captures how musical composition is distributed spectrally. Features relevant to Timbre are:

   a) **Spectral Centroid:** This gives the expected value of the spectral distribution of a frame. Large centroid reflect a bias towards higher frequencies and small reflect a bias towards lower frequencies.

   $$C_t = \frac{\sum_{n=1}^{N} M_t[n] * n}{\sum_{n=1}^{N} M_t[n]}$$

   where $M_t[n]$ is the magnitude of the Fourier transform at frame t and frequency bin n. The centroid is a measure of spectral shape and higher centroid values correspond to âĂIJbrighterâĂİ textures with more high frequencies.

   b) **Spectral Rollof:** This is the frequency below which 85% of the spectrum magnitude distribution is concentrated.

   $$\sum_{n=1}^{R_t} M_t[n] = 0.85 * \sum_{n=1}^{N} M_t[n].$$

   c) **Zero Crossing Rate:** It represents the number of times the waveform crosses zero in a window frame.

   $$Z_t = \frac{1}{2} \sum_{n=1}^{N} |sign(x[n]) - sign(x[n-1])|$$

   where the $sign$ function is 1 for positive arguments and 0 for negative arguments and $x[n]$ is the time domain signal for frame t.

   d) **Mel-Frequency Cepstral Coefficients:** It measures the repetition and predicity of frequency.

   e) **Chroma Frequencies:** It is the representation for music audio in which entire spectrum is projected onto 12 bins which represent 12 distinct semitones.

2. **Rhythm Features:** This extract information on the timing, beat, and tempo of the song and measures speed of the music. **Different Rhythm Features are:**

a) **First Tempo BPM:** Beats Per Minute of the strongest Tempo.
b) **Second Tempo BPM:** Beats Per Minute of the second strongest tempo.
c) **First Tempo Strength:** Strength of the strongest tempo.
d) **Second Tempo Strength:** Strength of the second strongest tempo.
e) **Estimated Tempo:** Estimated Tempo using Log-Normal Weighting.

4.3 DIFFERENT APPROACHES

1. **Preprocessing**

To improve accuracy of the system we have tried different approaches in pre-processing of the dataset.

Unsuccessful Attempt:
Since, the GIZAN dataset has 100 songs in each genre considering the classification done on 10 genres, there were total of 1000 songs. We tried extending the number of features from 34 to 66.

Extended features:

a) Chroma frequency mean : 12
b) Chroma frequency variance : 12
c) Mel mean : +2
d) Mel variance : +2
e) Mfcc mean : +2
f) Mfcc varience : +2

After running the same, we found that the accuracy decreased. Possible reason may be that the dataset was small and features was respectively more. Therefore, the training is not being done approximately.

Successful Attempt:
Further, we have tried decresing the features to a number which significantly increases the training and test accuracy. Output after such a change was much better with improved accuracy so we sticked with this approaches and eliminate the other useless features.

5 EXPERIMENTS:

5.1 CODE DESCRIPTION

1. **Language and Environment:** we have build this system with Python and Jupyter-notebook.
2. **Python library used:** Numpy, Pandas, Scipy, Scikt-learn, Librosa
3. **LoC:** This project include 200 line of codes.
4. **URL:** `https://github.com/anshulgupta0803/music-genre-classification`

5.2 EXPERIMENT PLATFORM

We have run our project code on our own laptop having specification:

1. Processor : Intel i5 5th gen
2. Ram : 8gb

Time Taken by different part of Code:

1. Pre-processing : 20 minutes
2. Training : 5 minutes

5.3 EXPERIMENTAL RESULTS

Classifier and hyper parameter

Table 5.1: Classifier and its hyper parameter

	Classifier	**Hyperparameter**
1.	Decision tree	Minimum sample split : 20
2.	Random forest	Minimum sample split : 10
3.	KNN	Default : 5
4.	Logistic regression	Max-iteration : 15000
5.	Linear SVM	Max-iteration : 10000
6.	Naive Bayes Bernoulli	Defalut
7.	Nave Bayes Gaussion	Default

Result for different combination of features

Table 5.2: Mapping of #Table to combination of features

	8 features	**contrast**	**Mfcc-coeff**	**Mfcc-mel**	**Chroma**	**Rythm**
table-3.	✓	X	7	X	X	✓
table-6.	✓	✓	5	5	✓	X
table-2.	✓	X	X	X	X	✓
table-1.	✓	✓	4	4	X	✓
table-4.	✓	✓	5	5	✓	✓
table-5.	X	✓	5	5	6	✓

Table 5.3: Table-1

	Train Accuracy	**Test Accuracy**
Decision Tree	74.28	40.5
Random Forest	94	57
KNN	48.31	26.5
Logistic Regression	56.55	49
Support Vector	22.97	25
Naive Bayes Bernoulli	17.97	15.5
Naive Bayes Gaussian	45.19	43

Table 5.4: Table-2

	Train Accuracy	**Test Accuracy**
Decision Tree	69.16	41
Random Forest	86.392	49.5
KNN	46.9413	27.5
Logistic Regression	37.578	30
Support Vector	15.8552	15.5
Naive Bayes Bernoulli	10.7366	7
Naive Bayes Gaussian	38.2022	33.5

Table 5.5: Table-3

	Train Accuracy	**Test Accuracy**
Decision Tree	71.78	40.5
Random Forest	91.63	60
KNN	47.44	24
Logistic Regression	39.70	41.5
Support Vector	23.97	21.5
Naive Bayes Bernoulli	29.96	29
Naive Bayes Gaussian	37.07	39.5

Table 5.6: Table-4

	Train Accuracy	**Test Accuracy**
Decision Tree	75.47	44
Random Forest	94.11	58
KNN	47.83	27
Logistic Regression	57.82	53
Support Vector	18.86	22
Naive Bayes Bernoulli	20.53	23
Naive Bayes Gaussian	45.72	44

Table 5.7: Table-5

	Train Accuracy	**Test Accuracy**
Decision Tree	76.35	45
Random Forest	93.45	62
KNN	63.15	43
Logistic Regression	67.70	57
Support Vector	47.39	41
Naive Bayes Bernoulli	19.97	24
Naive Bayes Gaussian	54.16	46

Table 5.8: Table-6

	Train Accuracy	**Test Accuracy**
Decision Tree	75.90	47
Random Forest	94.86	58
KNN	49.31	25
Logistic Regression	57.30	51
Support Vector	22.09	22
Naive Bayes Bernoulli	21.09	18.5
Naive Bayes Gaussian	46.44	39

6 Effort

Time we have spent in different parts of the project:

1. Preprocessing : 20 hrs
2. Training : 5 hours

Challenge we have faced:
Deciding which feature to select and which feature to drop. When we include some feature we found that for some classifier accuracy increases for other accuracy decreases.

Work allocation:

1. **Rajershi**< 193059002> **:** Literature Review and choosing the feautures for the classfiers.
2. **Yadnesh** < 193050067> **:** Understanding the various features of GTZAN dataset and using the feautures for classification with Aman.
3. **Aman** < 193050022> **:** Report, trying the various classifiers on the feautures.
4. **Suraj** < 193050003> **:** Discussions, Report, and presentation.

7 REFERENCES

[1] Musical Genre Classification of Audio Signals by George Tzanetakis Student Member IEEE and Perry Cook, Member IEEE.
[2] An article on medium by Parul Pandey :
https://towardsdatascience.com/music-genre-classification-with-python-c714d032f0d8
[3] Omar Diab, Anthony Manero, and Reid Watson. Musical Genre Tag Classification With Curated and Crowdsourced Datasets. Stanford University, Computer Science, 1 edition, 2012.
[4] N. Scaringella, G. Zoia, and D. Mlynek. Automatic genre classification of music content: a survey. IEEE Signal Process. Mag., 23(2):133âĂŞ141, 2006.