[[1]](#footnote-1)

Music Genres Recognition with Hidden Markov Models

Induja Vand Meghana Kantharaj

*Abstract*—Music has been an integral part of man’s life, and in the study of mankind, study of music history leads to acute explanation of the progression of human behavior through generations. Thus study of music becomes important. Study of music includes its origin, progression and transformation, influences and effects on the world in general. Different influences in music have given rise to a myriad number of genres, each with its own distinct features. Study of genres help in decoding history and also, the present. When technology is used as the medium, this problem becomes a data mining problem with machine learning application, which results in the discovery of knowledge. This project is a music genre identification framework that uses supervised machine learning to recognize the genre of a sample. It uses digital media processing to process the audio clipping and Hidden Markov Models for classification.

# INTRODUCTION

m

usic is an art form whose medium is sound. Its common elements are pitch, dynamics and the sonic qualities of timbre and texture. Combinations of these result in different genres of music. Every day, hundreds of new songs/ audio clippings are made available to the people. In due course of time, the songs database has become so large that it has become difficult to manage them. It becomes harder every day for people to manage their music libraries, as content grows.

There are a wide variety of factors that determine why a certain person may like a certain genre of music and may not like some other type of music. This gives rise to the need of genre based classification of the musical database at the service provider’s end so as to attract more customers to use their product by making their job simple.

Hence, there is a need to classify songs based on the kind of music it is, i.e. based on genres. This project Music Genre Recognition helps solve the issue by performing recognition.

Music Genre Recognition is a three step process-

1. Preprocessing
2. Training
3. Testing

Preprocessing is the first stage in music genre detection. It involves the collection of dataset of suitable form. Here audio files of the format .au have been collected. Then this data has to be preprocessed for erroneous data by performing suitable actions

Training stage involves the training of the classifier for classification of music samples. The various techniques that can be used for this task is Hidden Markov Model, Support Vector Machines, Neural Networks, Naïve Bayes classifier, etc. These are Machine Learning algorithms for supervised learning. The efficiency varies for each model.

Testing is the stage where inputs are tested for their genres to be detected. They are evaluated against the classes for the maximum probability and a class with the maximum probability is declared as its class. Here, to calculate the efficiency of the classifier, a confusion matrix is plotted. Confusion matrix contains information regarding false positives, true positives, true negatives and false negative. The efficiency can be calculate by dividing True Positive by total test cases.

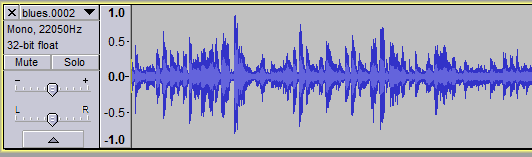
# data collection

Data collection for this project was sourced from online repositories. The data set called GTZAN. This dataset was used for the well-known paper in genre classification "Musical genre classification of audio signals” by G. Tzanetakis and P. Cook in IEEE Transactions on Audio and Speech Processing 2002.

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 .au format.

# preprocessing

The preprocessing stage for this project involved filtering the dataset for audio files of 4 genres-blues, classical, jazz and rock. Then the audio files were converted from .au to .wav format using the preprocessing module. Python was used in this module which used sunau and wav libraries. The visualization of the .wav samples were done using the tool Audacity. The sample rate of the file and the size were altered to a suitable value.



# FEATURE Extraction

The kind of features considered for the classification of music for this project is the Mel Frequency Cepstral Coefficients. In sound processing, the Mel-frequency cepstrum (MFC) is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear Mel scale of frequency. Mel-frequency cepstral coefficients (MFCCs) are coefficients that collectively make up an MFC. They are derived from a type of cepstral representation of the audio clip, a nonlinear "spectrum-of-a-spectrum”. For this, the python\_speech\_features library by Jameslyons from the GitHub repository was used. This library is used to generate normalized coefficients for an audio sample. Hamming window size of 30ms was applied on the .wav sample files used for training along with the window step attribute set to 30ms. The features of audio samples ranged from 1 to 20. These features were normalized to a range of 1 to 7. The features were extracted, normalized and written into different files, one for each genre. Each genre had a feature set containing 260000 members.

# Training

Training was done using the generated normalized feature vectors, however a smaller subset was considered for the training, due to large computation time setbacks. Hidden Markov Models were implemented in python and were used as the classifier. Four instances of HMMs were used, one corresponding to each genre of music. Each of these instances were trained with the corresponding genre feature vector. The training of the HMM took approximately 12 minutes. When the training stage is complete, the instance of the HMM is capable of generating the probability of data sample belonging to its class.

# Testing

Features of an audio sample are generated and the subset of generated feature vectors are considered for testing. The ratio of training to testing of the dataset is 70:30. These feature vectors are input to the instances of HMM, each of which generate a probability of the audio sample belonging to the category. The probabilities are compared and the HMM instance that gives the maximum probability for an audio sample is considered as the class of the audio sample. There are possibilities of the audio samples being wrongly classified as the considered dataset is small and the training done is inadequate.

# results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Blues | Classical | Jazz | Rock |
| Blues | 122 | 30 | 6 | 42 |
| Classical | 35 | 123 | 42 | 0 |
| Jazz | 12 | 92 | 96 | 0 |
| Rock | 84 | 13 | 3 | 100 |

*Confusion Matrix*

A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known. The confusion matrix is displayed for a sample set of 200 feature vectors of each genre.

The rows contain number of actual class classification and the columns contain the number of predicted class classification. The efficiency of the HMM, class-wise, is given by the number of True Positives divided by the total samples of that class. The overall efficiency is the average efficiency considering all classes.

# conclusion

The HMMs classified the audio samples successfully into their respective genres with an overall efficiency of 72% for a standard dataset and 55% for a smaller dataset.

The dataset contains two sets of music genres with musical similarities ie.., Blues and Rock, Jazz and Classical, due to which the efficiency is reduced when smaller dataset is considered

References

*Research Papers :*

[1] Molau, Pitz, Schlueter, Ney :Computing Mel- Frequency Cepstral coefficients on the Power Spectrum, RWTH Aachen – University of Technology, 52056 Aachen, Germany. IEEE Transactions on Communications 26 (6)

[2] Frigo, M.; Johnson, S. G. (2005). "The Design and Implementation of FFTW3" (PDF). Proceedings of the IEEE 93: 216–231. 10.1109/jproc.2004.840301

[3] Narasimha, M.; Peterson, A. (June 1978). "On the Computation of the

Discrete Cosine Transform". IEEE Transactions on Communications

26(6): 934–936. 10.1109/TCOM.1978.1094144

[4] S. B. Kotsiantis : Supervised Machine Learning: A Review of

Classification Techniques, 2007, University of Peloponnesse, Greece

[5] David A. Freedman (2009). Statistical Models: Theory and Practice.

Cambridge University Press. p. 128.

[6] Musical genre classification of audio signals, Speech and Audio

Processing, IEEE Transactions on (Volume:10 , Issue: 5 ), 293-302,

November 2002,

*Web Links:*

1. <http://willdrevo.com/fingerprinting-and-audio-recognition-with-python/>
2. http://en.wikipedia.org/wiki/Hidden\_Markov\_Model
3. <http://en.wikipedia.org/wiki/Mel_Frequency_Ceptral>
4. http://en.wikipedia.org/wiki/Window\_function

1. [↑](#footnote-ref-1)